11/12/2017



Ouverture automatique d’une porte et mise en marche d’un ventilateur

Sommaire :

1. Introduction
2. Objectif
3. Cahier de Charges
4. Choix de plateforme de prototypage
5. Montage
6. Code
7. Vidéo Démonstration
8. Conclusion

1. Introduction

Dans le cadre du Projet de deuxième année cycle ingénieur, Abdelkarim Chaara, et moi-même, avons décidés de choisir un sujet qui nous semblait intéressant : « La programmation pour le contrôle d’une partie de la maison ». Ou plus précisément l’ouverture de la porte d’entrée déclenché par un détecteur de mouvement et la mise en marche d’un ventilateur quand la température augmente.

Ce qui nous a attirés dans ce projet réside dans le fait que c’est une technologie récente mais qui se développe rapidement. De même nous avons vu de multiples avantages à approfondir ce projet, en effet, cela nous permet de nous améliorer dans le domaine de la programmation mais aussi car cette technologie est irrévocablement liée à l’informatique, notre passion, et bien évidemment car elle se développe de plus en plus chez les particuliers, et donc pourquoi pas, bientôt chez nous. En effet la domotique présente plusieurs avantages indéniables, elle permet d’assurer la sécurité de la maison de par ses multiples capteurs, caméras et autres appareils électroniques. Elle permet aussi d’être plus écologique grâce à sa gestion de la température ambiante. Et enfin l’avantage le plus important est que la domotique permet d’être mieux assisté au quotidien (notamment pour les personnes handicapées)

C’est pour ces multiples raisons que notre choix fut unanime et enthousiaste concernant ce  
projet de programmation, malgré le fait que nous ne connaissions que très peu de choses sur la domotique avant d’entamer nos recherches documentaires.

1. Objectif

Le thème de notre projet est La programmation pour le contrôle d’une partie de la maison. Ou plus précisément l’ouverture de la porte d’entrée déclenché par un détecteur de mouvement et la mise en marche d’un ventilateur quand la température augmente.

Concernant notre projet de programmation pour le contrôle de la maison nous avions, dans la première partie, cherché des informations concernant Tous les composants qu’on va utiliser et comment réaliser le montage adéquat. Puis, suite à l'achat d'une carte  
Arduino, nous avons réalisé l'ensemble de nos programmes sous l'environnement de développement Arduino afin de faciliter le transfert de nos programmes sur la carte.

1. Cahier de Charges

Les matériels nécessaires pour la réalisation du projet :

* **Carte Arduino UNO :**

Une carte Arduino est une petite (5,33 x 6,85 cm) carte électronique équipée d'un microcontrôleur.  
Le microcontrôleur permet, à partir d'événements détectés par des capteurs, de programmer et commander des actionneurs ; la carte Arduino est donc une interface programmable.

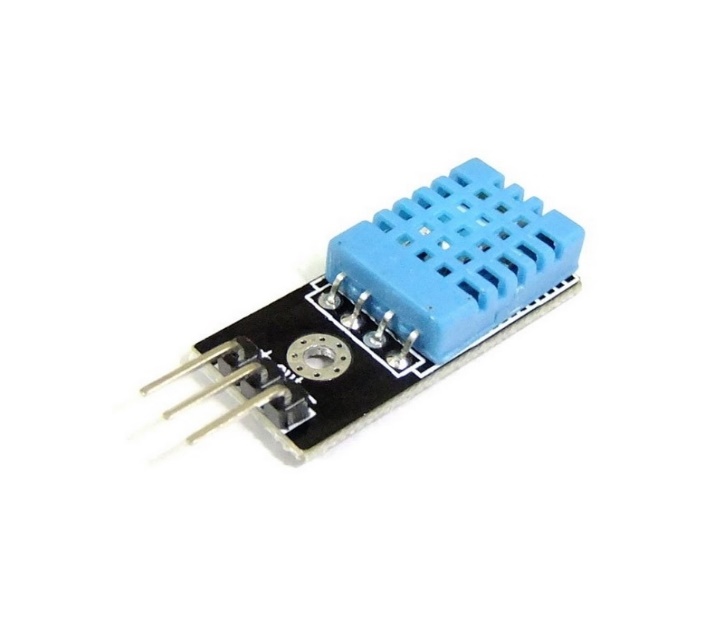
* **Détecteur de mouvement PIR :**

Les capteurs PIR vous permettent de détecter les mouvements, presque toujours utilisés pour détecter si un humain est entré ou sorti de la plage des capteurs. Ils sont petits, peu coûteux, de faible puissance, faciles à utiliser et ne s'usent pas. Pour cette raison, ils se trouvent généralement dans les appareils et les gadgets utilisés dans les maisons ou les entreprises. Ils sont souvent appelés capteurs PIR, "Passive Infrared", "Pyroélectrique" ou "IR Motion".



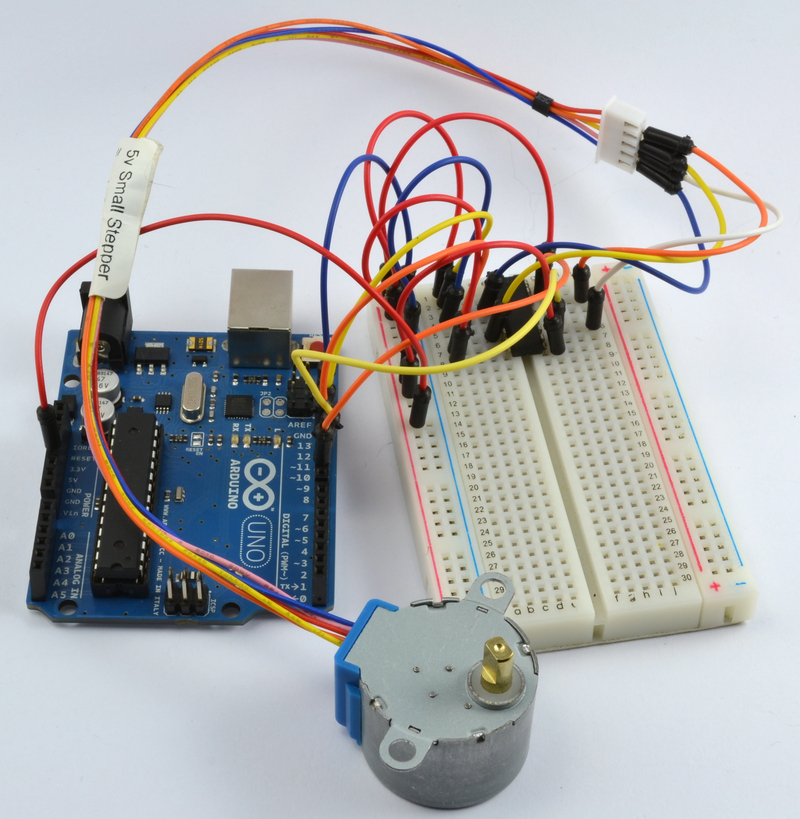
* **Capteur humidité / température "DHT-11" :**

Le DHT-22 est un capteur d'humidité et de température bénéficiant d'une interface 1 Wire™. Livré pré-calibré, ce dernier ne nécessite aucun autre composant.

**Caractéristiques :**- Alimentation : 3,3 à 6 V  
- Consommation : 1 à 1,5 mA  
- 40 à 50 uA en stand-by  
  
- Humidité : 0 - 100 % RH  
- Précision : +/- 2% RH  
   
- Température : -40 à 80 °C  
- Précision : +/- 0,5 °C

* **DC motor :**

Le moteur DC de votre kit Arduino est le plus basique des moteurs électriques et est utilisé dans tous les types d'électronique de loisir. Lorsque le courant est traversé, il tourne continuellement dans une direction jusqu'à ce que le courant s'arrête. Sauf indication contraire de + ou -, les moteurs CC n'ont pas de polarité, ce qui signifie que vous pouvez inverser les deux fils pour inverser la direction du moteur.

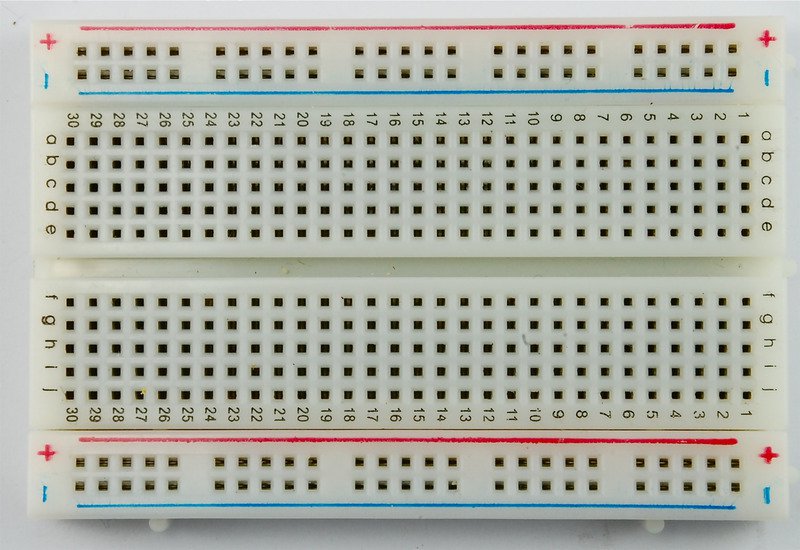


* **Servo moteur :**

Un servomoteur (souvent abrégé en « Servo », provenant du latin servus qui signifie « esclave ») est un [moteur](https://fr.wikipedia.org/wiki/Moteur) capable de maintenir une opposition à un effort statique et dont la position est vérifiée en continu et corrigée en fonction de la mesure. C'est donc un système asservi. Le servomoteur intègre dans un même boitier, la mécanique ([moteur](https://fr.wikipedia.org/wiki/Machine_%C3%A0_courant_continu) et engrenage), et l’électronique, pour la commande et l'asservissement du moteur. La position est définie avec une limite de débattement d’angle de 180 degrés, mais également disponible en rotation continue

* **Breadboard :**

Une platine d'expérimentation ou platine de prototypage (appelée en anglais breadboard ou encore Labdec du nom de la marque la plus répandue) est un dispositif qui permet de réaliser le [prototype](https://fr.wikipedia.org/wiki/Prototype) d'un [circuit électronique](https://fr.wikipedia.org/wiki/Circuit_%C3%A9lectronique) et de le tester. L'avantage de ce système est d'être totalement réutilisable, car il ne nécessite pas de [soudure](https://fr.wikipedia.org/wiki/Soudage).

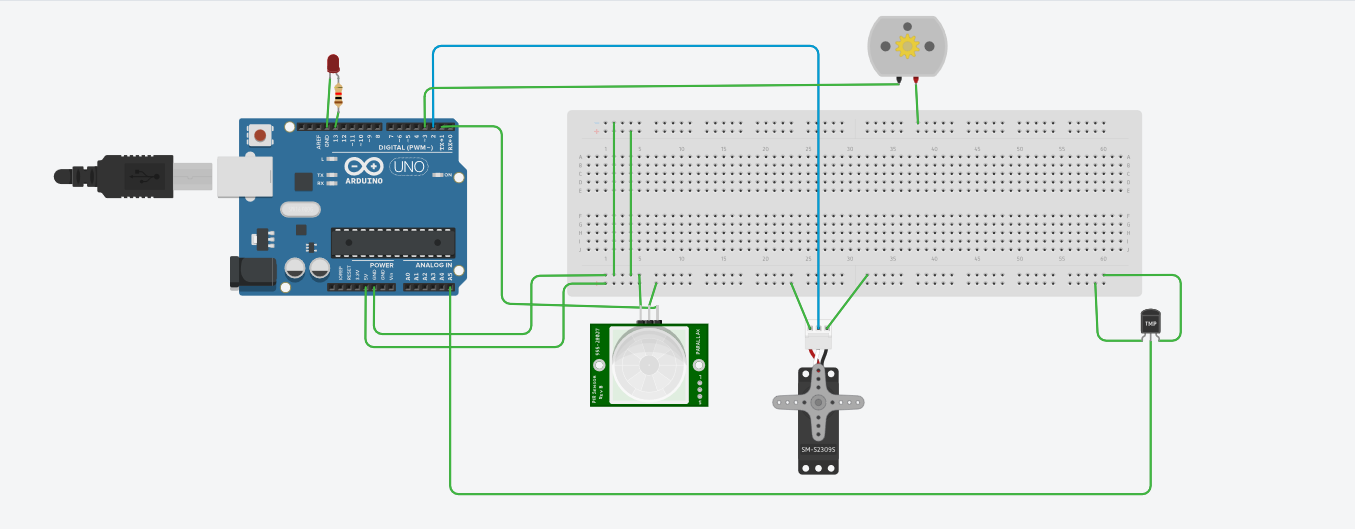


1. Choix de plateforme de prototypage

Pour le choix de la plateforme de prototypage, nous avons opté pour Circuit.io vu qu’on s’est déjà familiariser avec l’année précédente. Et en plus de ça c’est une plateforme gratuite et pratique pour la réalisation des différents circuits nécessaires. Elle contient aussi tous les composants dont on aura besoin pour la réalisation de notre projet



1. Montage



1. Code Commenté

#include <Servo.h>

/\*

 \* PIR sensor tester

 \*/

**int** pos = 0;

Servo s ;

**int** led =13;

**int** inputPin = 1; // choose the input pin (for PIR sensor)

**int** pirState = LOW; // we start, assuming no motion detected

**int** val = 0; // variable for reading the pin status

**int** vt=0; // variable for reading TEMP

**int** servo\_pin=2;

**int** dc\_pin=3;

**int** temp\_pin=5;

**void** setup() {

pinMode(led, OUTPUT); // declare LED as output

pinMode(dc\_pin, OUTPUT); // declare DC as output

pinMode(inputPin, INPUT); // declare sensor as input

pinMode(temp\_pin, INPUT);

s.attach(servo\_pin);

Serial.begin(9600);

}

**void** loop(){

vt = analogRead(temp\_pin);

**float** mv = (vt/1024.0)\*5000;

**float** cel = (mv/10)-49;

**float** farh = (cel\*9)/5+32;

Serial.println("temperature: ");

Serial.println(cel);

**if**(cel>30){

digitalWrite(dc\_pin,1);

Serial.println("Ventilateur: ON");

} **else** {

digitalWrite(dc\_pin,0);

Serial.println("Ventilateur: OFF");

}

val = digitalRead(inputPin); // read input value

Serial.println(val);

**if** (val == HIGH) { // check if the input is HIGH

//map(value, fromLow, fromHigh, toLow, toHigh)

**if** (pirState == LOW) {

// we have just turned on

Serial.println("Motion detected!");

Serial.println("OPEN GATE ");

digitalWrite(led, HIGH);

**for** (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees

// in steps of 1 degree

s.write(pos); // tell servo to go to position in variable 'pos'

delay(10); // waits 15ms for the servo to reach the position

}

pirState = HIGH;

}

}

**else** {

digitalWrite(led, LOW); // turn LED OFF

**if** (pirState == HIGH){

// we have just turned of

Serial.println("Motion ended!");

**for** (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0 degrees

s.write(pos); // tell servo to go to position in variable 'pos'

delay(10); // waits 15ms for the servo to reach the position

}

Serial.println("Closed GATE");

// We only want to print on the output change, not state

pirState = LOW;

}

}

}

1. Vidéo Démonstration



Figure 1 Démonstration vidéo

1. Conclusion

Nous n'avons pas pu réaliser un produit commercialisable et nous n'avons pas eu le temp pour le concrétiser avec du vrai matériel, mais nous avons tout de même réussi à remplir presque l'ensemble de nos objectifs.

Ce projet nous a permis de voir que grâce à nos connaissances que nous avons  
acquises cette année nous n'arrivons certes pas à finaliser notre idée complètement mais au  
moins de comprendre ce que nous avons appris en l'appliquant et en créant des petits objets  
connectés par nous-mêmes, de chercher des solutions aux problèmes et d'en voir les résultats.  
C’est une façon plus amusante de voir et d’apprendre l’électronique.  
Nous avons été assez contentes et satisfaites de notre résultat final. De plus, notre idée n’est pas encore sur le marché. Qui sait, peut-être qu’un jour elle sera aboutie.