

→ les étapes de réalisation d'un cahier des charges

→ tout d'abord on précise le cahier des charges

→ Préciser les entrées et sorties

→ Circuit de commande

Phase 1: Elaboration du programme

Phase 2: Exécution du programme dans la carte ARDUINO

→ Matériel.

→ Définition de ARDUINO

→ ARDUINO est une plateforme électronique basée sur du matériel et des logiciels simples à utiliser. Arduino est constituée d'une carte électronique et d'un environnement de programmation.

→ La carte Arduino est l'endroit où le code que vous écrivez est exécuté.

→ Exemple (Entrées)
Exemple de capteurs :

→ Bouton poussoir

→ Capteur de température

→ Capteur de lumière

→ Potentiomètre

Exemple d'Actionneurs :
(sorties)

→ LED

→ Moteur

→ Buzzier

→ Pourquoi ARDUINO?

- le logiciel ARDUINO est facile à utiliser et assez flexible pour les utilisateurs de réaliser des projets à moindre coût.
- le logiciel ARDUINO fonctionne sur les systèmes d'exploitation windows, MAC OS et linux
 - Environnement de programmation simple et clair
 - Nombreuses bibliothèques disponibles avec diverses fonctions implémentées.

Prix Arduino Uno: 18€

Logiciel: 0€

Assistance: Forum Arduino

→ Quelle est la différence entre la CARTE ARDUINO MEGA et UNO?

- la carte Arduino Mega est plus grande et a plus d'options de connexion et des processeurs plus puissants
- la carte Uno est souvent préférée pour des projets simples car elle a un coût inférieur.
- les 3 types de mémoires qui coexistent au sein d'un Arduino Uno à base ATmega 328 sont:
- la mémoire FLASH.
- ~ SRAM (Static Read Access memory)
- ~ EEPROM (Electrically Erasable Programmable Read Only Memory)

(8) Broche 5V et GND : Elles fournissent 5V et la masse au circuit réalisé.

(9) Les pins d'entrées analogiques : Elles sont utilisées pour mesurer une tension variable (entre 0 et 5V) qui peut provenir des capteurs ou d'interfaces diverses (potentiomètre, etc...)

→ Programmer avec ARDUINO

→ le langage ARDUINO se base sur le C/C++.

* La structure du programme Arduino

(1) Déclaration des variables

(2) void setup() : cette partie est dédiée à l'entrée/borne.

(3) void loop() : Elle contient le programme principal.

* Les fonctions d'entrée/sorties :

pinMode (pin, Mode) : Elle sert à configurer une pin numérique pour se comporter soit comme entrée ou sortie. Exemple : pinMode (7, INPUT)

digitalWrite (pin, valeur) : Avec cette fonction, la pin numérique choisie prend l'état ON (valeur 1) ou l'état OFF (valeur 0).
Exemple : digitalWrite (8, HIGH)

digitalRead (pin) :

Elle lit l'état d'une pin, elle retourne 1 ou 0. Exemple : val = digitalRead (7)

analogRead (pin) :

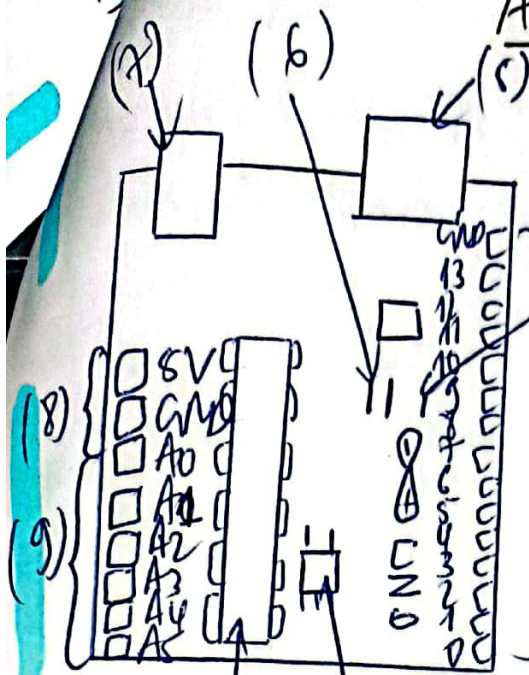
Elle lit la tension appliquée à une entrée analogique.

Elle retourne une valeur entre 0 et 1023 qui correspond à une valeur entre 0 et 5V. Exemple : val = analogRead (0) ;

analogWrite (pin, valeur) :

cette fonction permet d'envoyer une valeur analogique à la pin.
Elle peut être 3,5, 9, 6, 10, 11.

Anatomie d'ARDUINO



(1) Microcontrôleur Atmega,
Il représente le cœur de la carte Arduino

(2) Bouton de remise à zéro, Il permet de remettre à zéro le micro contrôleur.

(3) Les pins d'entrée / sorties numériques (0-13), les pins peuvent être utilisés pour recevoir ou envoyer des signaux numériques "0" ou "1".

(3) Les pins de sorties analogiques (3, 5, 6, 9, 10 et 11), ce sont des pins numériques qui peuvent être programmés pour délivrer des signaux analogiques

(4) LED, Elle est soudée sur la carte ARDUINO et reliée à la pin 13.

(5) Port USB, Il sert à alimenter la carte Arduino, transférer le programme et établir la communication entre Arduino et le moniteur série

(6) LED de transmission et réception, Elles indiquent la communication entre la carte ARDUINO et l'ordinateur.

(7) Connecteur d'alimentation, Il permet de connecter une alimentation à la carte ARDUINO quand le port USB n'est pas utilisé. le connecteur accepte une tension entre 5V-12V.

Pour changer le rapport cyclique du PWM. la valeur peut être entre 0 et 255 ce qui représente une valeur dans l'intervalle 0 à 255.
Exemple: analogWrite (9, 128);

* les fonctions de temps

delay (valeur) : Mettre une pause au programme (en ms)
delayMicroseconds (valeur) : Mettre une pause au programme (en µs)

* les fonctions de communication

Serial.begin (débit);

cette fonction sert à fixer le débit de communication. Pour communiquer avec l'ordinateur, le débit 9600 bauds est généralement utilisé

Exemple: Serial.begin (9600);

Serial.available();

elle donne le nombre de caractères disponible pour lecture dans la file d'attente (buffer) du port série. Elle renvoie 0 si aucun caractère n'est disponible. Sa valeur sera supérieur à 0 quand une donnée arrive. Exemple: int x = Serial.available();

Serial.read();

elle lit des données entrantes sur le port série

Exemple: int data = Serial.read();

Exercice

Ecrire un programme qui allume une led 1000ms et éteindre

int led = 5;

void setup() {

pinMode (led, OUTPUT);

}
void loop() {

digitalWrite (led, HIGH);