

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement et de la Recherche Scientifique

Université Abou bekr BELKAID Tlemcen

Faculté des Sciences
Département d'Informatique



Mémoire de Master en Informatique
Option : **Modèle intelligent et Décision**
Sujet :

**Conception et réalisation d'un prototype IoT
intelligent basé sur la logique floue pour la détection
du monoxyde de carbone (CO)**

Présenté par : **MEDDAH Youcef**
MOULAI KHATIR Aïssam

Date : 15/07/2021.

Membres du Jury :

Président	Dr. BENAMMAR Abd el Karim	Université de Tlemcen
Encadrant	Dr. MERZOUG Mohammed	Université de Tlemcen
Co-Encadrant	Dr. HADJILA Mourad	Université de Tlemcen
Examineur	Dr. BELABED Amine	Université de Tlemcen

Année Universitaire 2020/2021

Résumé

Le monoxyde de carbone constitue la cause d'intoxications domestiques extrêmement fréquentes, parfois mortelles. Le bilan macabre que fait le CO itératif à chaque saison hivernale, est souvent dû à l'utilisation des appareils de chauffage et chauffe-bains. Le CO est un gaz inodore, incolore et non irritant, ce qui rend sa détection très difficile, en l'absence d'un détecteur. Pour faire face à ce fléau mortel, et sauver les vies humaines, nous avons pensé à concevoir et réaliser un prototype à base d'internet d'objets. Ce dernier permet dans un premier temps de détecter les fuites liées au monoxyde de carbone, et dans un second temps d'apporter les solutions envisageables en utilisant la logique floue. Ces solutions permettent d'activer d'abord un système de ventilation afin d'aérer l'endroit où se produit la fuite, ensuite, d'avertir des personnes en leur envoyant des alertes sous forme d'appels téléphoniques. Concrètement, ce travail est effectué à l'aide d'un capteur VOC, calibré pour détecter le niveau de gas présent dans l'air, et associé à la carte de développement arduino pour alerter les utilisateurs lorsque ce niveau dépasse un certain seuil.

Mots-clés : capteurs VOC, module GSM, CO, arduino, internet d'objets.

Abstract

Carbon monoxide is the cause of extremely frequent household poisoning, sometimes fatal, the fatalities that the iterative CO makes in each winter season, is often due to the use of heating devices and bath heaters. CO is an odorless, colorless, non-irritating gas, which makes it very difficult to detect without a detector. To face this deadly scourge, and save human lives, we have thought of designing and producing a prototype based on the Internet of Things, the latter first of all making it possible to detect carbon monoxide-related leaks, and secondly, to provide possible solutions using fuzzy logic. These solutions allow a ventilation system to be activated, first to ventilate the area where the leak occurs, and then to alert those present by sending them alerts in the form of cellphone calls. Concretely, this work is carried out using a VOC sensor, calibrated to detect the level of gas present in the air, and associated with the arduino development board to alert users when this level exceeds a certain threshold.

Key-Words : VOC sensors, GSM module, CO, arduino, Internet of things.

Dédicaces

Je dédie ce modeste travail
A mes très chers parents.
A mes frères.

Remerciements

Je remercie ALLAH, le tout puissant, le miséricordieux, de m'avoir donné le courage, la volonté et la patience de mener à terme ce présent travail.

Je veux adresser mes remerciements à mon encadreur Mr. **MERZOUG Mohammed**, plus personnellement pour avoir d'abord proposé ce thème et le suivi continué tout le long de la réalisation de ce travail et qui n'a pas cessé de me donner ses conseils.

Je remercie aussi tout les enseignants de la spécialité informatique qui ont contribué à ma formation.

Je remercie tous les membres de jury pour la pertinence de leurs remarques.

Je remercie mes très chers parents et frère qui ont toujours été là pour moi.

MOULAI KHATIR Aïssam

Dédicaces

Je dédie ce modeste travail

A ceux qu'ont faits de moi un homme, mes très chers parents " mon père " et " ma mère ".

A ma sœur.

A Mes frères.

A toute ma famille.

A mes chers amis.

Remerciements

Je remercie ALLAH, le tout puissant, le miséricordieux, de m'avoir donné le courage, la volonté et la patience de mener à terme ce présent travail.

Je veux adresser mes remerciements à mon encadreur **Mr. MERZOUG Mohammed**, plus personnellement pour avoir d'abord proposé ce thème et le suivi continué tout le long de la réalisation de ce travail et qui n'a pas cessé de me donner ses conseils.

Je remercie tous les membres de jury pour la pertinence de leurs remarques.

Je remercie aussi tout les enseignants de la spécialité informatique qui ont contribué à ma formation.

Je remercie mes très chers parents et frère, mon uncle qui ont toujours été là pour moi.

MEDDAH Youcef

TABLE DES MATIÈRES

Table des matières	4
Table des figures	6
Liste des tableaux	8
Liste des Abréviations	10
Introduction Générale	11
1 Présentations : Théorie et Composants	13
1.1 Introduction	13
1.2 Asphyxie par Monoxyde de Carbone	13
1.2.1 Introduction	13
1.2.2 L'effet de concentration du CO sur une personne	14
1.3 La domotique	16
1.3.1 Qu'est-ce que la domotique ?	16
1.3.2 Les domaines de la domotique	16
1.4 Les Systèmes Embarqués	16
1.4.1 Qu'est ce qu'un système embarqué ?	16
1.4.2 Les caractéristiques d'un système embarqué [7]	17
1.4.3 Les avantages d'un système embarqué [7]	18
1.5 Présentation des Composants	18
1.5.1 Arduino Uno	18
1.5.2 Qu'est ce qu'un micro-contrôleur ?	18
1.5.3 Pourquoi arduino ? [8]	21
1.5.4 Câbles jumper et résistances	22

1.5.5	Module GSM SIM800L	23
1.5.6	Circuit de Mesure : Capteurs de Gas	24
1.5.7	Le Relai	31
1.5.8	Led, bi-led et buzzer	32
1.5.9	LCD I2C Crystal	33
1.5.10	Prototype Shield V5.0 et BreadBoard	33
1.6	Présentation de l'IDE Arduino	34
1.7	Principe de la logique floue	35
1.8	Comparaison avec des travaux antérieurs	37
1.9	Conclusion	37
2	Assemblage : Conception, Réalisation et Branchement	39
2.1	Introduction	39
2.2	Schéma principal et diagramme d'activité	39
2.2.1	Schéma principal	39
2.2.2	Diagramme d'activité	40
2.2.3	Niveaux d'alarmes	41
2.2.4	Application de la logique floue	43
2.3	Quelques parties du Code	45
2.3.1	Présentation des bibliothèques	45
2.3.2	Quelques fonctions et leur définition	46
2.4	Réalisation du Branchement	47
2.4.1	Composants nécessaires	47
2.4.2	Étape 1 : La carte arduino et le prototype shield V5.0	48
2.4.3	Étape 2 : Le Module MQ-2	49
2.4.4	Étape 3 : Les Leds et le Buzzer	49
2.4.5	Étape 4 : Le relai et la ventilation	51
2.4.6	Étape 5 : Le Module LCD I2C Crystal	51
2.4.7	Étape 6 : Le Module GSM SIM800L	52
2.4.8	Vue Globale des Branchements	55
	Conclusion et Perspectives	57
	Bibliographie	59

TABLE DES FIGURES

1.1	Statistique Asphyxie par Monoxyde (CO)	14
1.2	Some domains of embedded systems[9].	17
1.3	Atmega328p [12].	19
1.4	Pinout of the Arduino Uno [11].	20
1.5	Jumper Cables[13].	22
1.6	Resistances [24].	22
1.7	SIM800L GSM Module[14].	23
1.8	Pinout of GSM SIM800L Module[14].	24
1.9	MQ-2 Module[21].	26
1.10	MQ-5 Module[20].	26
1.11	MQ-7 Module[22].	27
1.12	MQ-9 Module[23].	27
1.13	Pinout of MQ-2 Module[10].	28
1.14	Sensivity's Characteristics of MQ-2 Module[10].	28
1.15	MQ2 Gas Sensor Parts[10].	29
1.16	Inside Gas Sensor Internal Structure of MQ-2 Sensor[10].	29
1.17	Sensing-Element[10].	30
1.18	Internal structure of sensing element[10].	30
1.19	Relay[19].	31
1.20	Buzzer [17].	32
1.21	Light-Emmiting Diode (LED) [16].	32
1.22	Bi-color Led [15].	32
1.23	LCD I2C Crystal Display	33
1.24	Prototype Shield and BreadBoard	33
1.25	Arduino IDE	34

1.26	Verify Button	34
1.27	Upload Button	34
1.28	Serial Monitor Button	35
1.29	Un système d'inférence floue	37
2.1	Principal Diagram	40
2.2	Activity Diagram	40
2.3	Stade 0	41
2.4	Stade 1	42
2.5	Stade 2	42
2.6	Stade 3	43
2.7	Fonctions d'appartenance de la variable d'entrée.	44
2.8	Fonctions d'appartenance de la variable de sortie.	44
2.9	Arduino and Prototype Shield V5.0 before connection	48
2.10	Arduino and Prototype Shield V5.0 after connection	48
2.11	MQ-2's Connection	49
2.12	Buzzer's Connection	50
2.13	Led's Connection	50
2.14	Relay and Ventilaton's connection	51
2.15	LCD I2C Crystal's connection	52
2.16	GSM SIM800L Module's connection	53
2.17	Full system after connection(without breadboard).	55
2.18	Full system after connection(with breadboard).	55
2.19	Le système après branchement (vue réelle)	56

LISTE DES TABLEAUX

1.1	The effects of CO concentration in the air [1].	15
1.2	Arduino Uno characteristics [3]	21
1.3	Tableau des comparaisons	37

LISTE DES ABRÉVIATIONS

Symbole	Signification
AT	ATtention
BPS	Bits Par Seconde
CO	Monoxyde de Carbone
CO2	Dioxyde de Carbone
COM	Common
CPU	Central Processing Unit
GND	Ground
GUI	Graphical User Interface
GSM	Global System for Mobile
GPRS	General Packet Radio Service
ICSP	In-Circuit Serial Programming
IDE	Integrated Développement Environnement
I2C	Inter-integrated Circuit
LCD	Liquid Crystal Display
LPG	Liquified Petroleum Gas
NC	Normally Closed
NO	Normally Open
PCB	Printed Circuit Board
PPM	Particule Par Million
PWM	Pulse Width Modulation
RTOS	Real Time Operating System
TWI	Two Wire Interface
TLV ou VLE	Threshold Limit Value ou Valeur Limite d'Exposition
UART	Universal Asynchronous Receiver-Transmitter
VCC	Voltage Collector Collector

INTRODUCTION GÉNÉRALE

Durant ces dernières décennies, plusieurs personnes ont trouvées la mort par asphyxie par monoxyde de carbone (CO) en Algérie surtout à l'approche de chaque saison hivernale. Plusieurs raisons sont à l'origine de cette catastrophe, dont on peut citer : la mauvaise aération, la contrefaçon, les mauvaises installations et le non-respect des normes d'entretien.

Il est nécessaire que des systèmes de sécurité efficaces soient installés dans des environnements domestiques, industriels et dans des établissements d'éducation.

Problématique

Un système d'alarme moderne doit impérativement comporter des détecteurs qui pourraient détecter tout événement pouvant mettre en danger la vie humaine telle que les incendies, les fuites de gaz, etc. Il doit aussi comporter des avertisseurs afin de signaler la production d'un événement. Quels sont les systèmes capables de prévenir ces dangers ? Peut-on produire un système autonome, efficace et à domicile afin de réduire ces risques ?

Objectifs

Le développement de la micro électronique et de la micro-informatique dans les circuits modernes notamment les micro contrôleurs sera exploité dans ce travail afin de concevoir un système économe embarqué basé sur le module Arduino et d'un capteur et d'autres composants, pour la mesure, la prévention et l'alerte en cas de fuite ou de présence de monoxyde de carbone, en minimisant les sources du problème et en alertant l'utilisateur d'une façon anticipée pour que sa réaction soit à temps et afin de prévenir ces risques.

Contributions

Ce travail a été réalisé dans le but :

- De contribuer à la protection de la vie humaine.
- Améliorer l'utilisation des systèmes embarqués et aider à leur utilisation.

Organisation du Travail

Dans le premier chapitre, nous donnerons une description générale des systèmes embarqués, ainsi que quelques statistiques sur les asphyxies par monoxyde en Algérie. Ce chapitre sera consacré à une étude approfondie des principaux équipements utilisés pour la mise en œuvre du système étudié où un modèle de base sera détaillé : sa construction, son environnement de programmation et son principe de fonctionnement. Nous présenterons aussi une étude de la famille des capteurs de gaz MQx pour la partie d'acquisition avec présentation des caractéristiques techniques relatives aux gammes de mesure.

Le deuxième chapitre représentera les différentes parties software et hardware pour le développement et la réalisation de l'application de mesure de concentration de monoxyde pour l'alerte en cas de dépassement du seuil toléré.

Enfin, nous terminons avec une conclusion générale et quelques perspectives.

CHAPITRE 1

PRÉSENTATIONS : THÉORIE ET COMPOSANTS

1.1 Introduction

Dans cette partie, nous allons définir quelques informations en ce qui concerne l'asphyxie par CO en Algérie, et donnerons des statistiques sur ce problème, nous verrons les conditions pour lesquels le CO devient un risque, nous présenterons les composants constituant le système et donnerons au détail les composants principaux du capteur de gas MQ-2, nous conclurons avec la présentation de l'ide Arduino et terminerons avec le principe de logique floue.

1.2 Asphyxie par Monoxyde de Carbone

1.2.1 Introduction

Le phénomène de décès par inhalation de gas a atteint des proportions alarmantes en Algérie. À l'approche de chaque saison hivernale, des centaines de citoyens trouvent la mort par asphyxie au monoxyde de carbone (CO). En 2009, 253 morts ont été déplorés par les services de la Protection civile. Il ne se passe pas une année sans que ce danger n'emporte avec lui des familles entières à travers les différentes régions du pays. Invisible, inodore et indétectable, ce gas s'incruste dans le sang à la place de l'oxygène et provoque la mort. Même conscientes, les personnes atteintes sont incapables de réagir dans la plupart des cas [5].

Les statistiques montrent que l'apparition de 0,1% de CO dans l'air tue en 1 heure, 1% en 15 minutes et 10% dans l'immédiat, c'est à dire l'ampleur du danger que représente ce gas lorsqu'il est libéré dans l'air. Quelques raisons majeures sont à l'origine des cas d'étouffement par le gas carbonique, dont on évoque la vétusté et la défectuosité du matériel de chauffage [5].

Le monoxyde de carbone est le résultat d'une mauvaise combustion quelle que soit la source d'énergie utilisée notamment le butane, l'essence, le charbon, le gas naturel, le pétrole, le propane, etc. Sa densité est voisine de l'air, Il se diffuse donc très vite dans l'environnement. Il existe, deux types d'intoxication. L'une est faible dite "chronique" et se manifeste à travers des maux de tête, des nausées, une confusion mentale, de la fatigue. Elle est lente et ses symptômes peuvent ne pas se manifester immédiatement. L'autre est aigüe, rapide et entraîne des vertiges, une perte de connaissance, une impotence musculaire, des troubles de comportement voire le coma ou le décès [5].

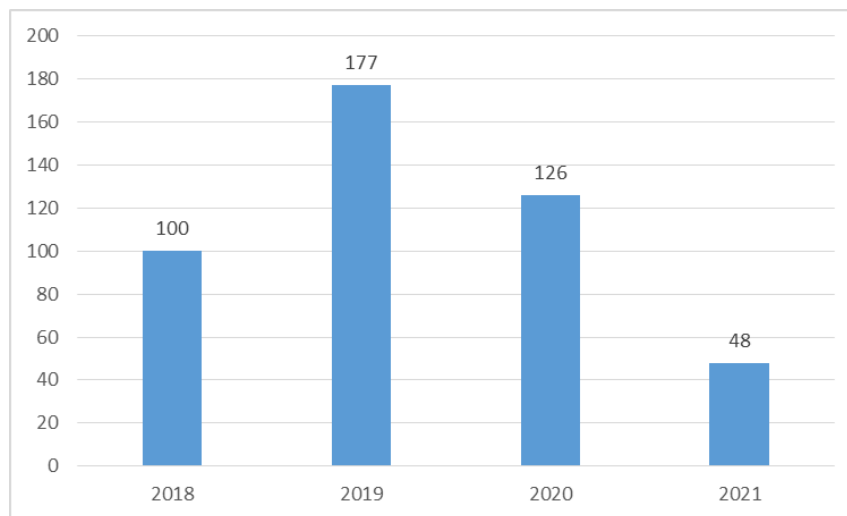


FIGURE 1.1: Statistique Asphyxie par Monoxyde (CO)

1.2.2 L'effet de concentration du CO sur une personne

Le taux habituel de CO dans l'air ambiant est d'environ 0,2 ppm (particule par million) [6].

L'Organisation mondiale de la Santé a défini, pour l'ensemble de la population, y compris les femmes enceintes et les personnes âgées atteintes d'affections cardiaques ou respiratoires (connues ou non), des valeurs de référence considérées comme inoffensives en fonction de la durée de l'exposition [6] :

- 10 mg/m^3 (9 ppm) pendant 8 heures.
- 30 mg/m^3 (26 ppm) pendant 1 heure.

- 60 mg/m^3 (52 ppm) pendant 30 min.
- 100 mg/m^3 (90 ppm) pendant 15 min.

La gravité d'une intoxication est fonction de la concentration de CO dans l'air et de la durée de l'exposition [1] :

CO (ppm)	% CO dans l'air	Symptômes
100	0.01	
200	0.02	Maux de tête, vertiges, nausées, fatigue
400	0.04	Maux de tête intenses, danger de mort après 3 heures.
800	0.08	Maux de tête, vertiges, nausées. Perte de connaissance en 45 min, décès après 2-3 heures.
1600	0.16	Symptômes sévères après 20 min, décès en moins d'une heure.
3200	0.32	Maux de tête, vertiges, nausées après 5 min, perte de connaissance après 30 min.
6400	0.64	Céphalées et vertiges après 1 à 2 min, perte de connaissance après 10-15 min.
12800	1.28	Perte de connaissance immédiate, décès en 1 à 3 minutes.

TABLE 1.1: The effects of CO concentration in the air [1].

La réglementation du travail définit des valeurs maximales de concentration auxquelles les travailleurs peuvent être exposés sans risques. Elles ne sont valables qu'en milieu de travail. Il faut garder à l'esprit qu'une quantité de CO tolérable pour un travailleur en bonne santé peut déjà représenter un certain danger pour une personne âgée ou pour un enfant [6].

La **TLV¹-TWA (Threshold Limit Value - Time Weighted Average)** représente la concentration moyenne considérée comme inoffensive pour un travailleur qui y est exposé 40 heures par semaine pendant toute sa carrière. Elle est de 20 ppm [6].

La **TLV-C (Threshold Limit Value - Ceiling)** représente la valeur plafond qui ne peut jamais être dépassée. Dans le cas du CO, la TLV-C est de 100 ppm [6].

1. TLV ou Valeur Limite d'Exposition(VLE) est une mesure du domaine de la pollution atmosphérique, de l'eau ou des sols, notamment utilisée dans les réglementations officielles.

La **TLV-IDHL** (**Threshold Limit Value - Immediately dangerous for life and health**) représente la concentration à partir de laquelle l'évacuation immédiate s'impose. Bien qu'une évacuation prenne en général moins de 30 minutes, pour des raisons de sécurité, les valeurs TLV-IDHL sont basées sur les effets d'une exposition durant 30 minutes. Elle est de 1500 ppm dans le cas du CO [6].

1.3 La domotique

1.3.1 Qu'est-ce que la domotique ?

La domotique est l'ensemble des techniques de l'électronique, de physique du bâtiment, d'automatisme, de l'informatique et des télécommunications utilisées dans les bâtiments, plus ou moins « interopérables » et permettant de centraliser le contrôle des différents systèmes et sous-systèmes de la maison et de l'entreprise (chauffage, volets roulants, porte de garage, portail d'entrée, prises électriques, etc.). La domotique vise à apporter des solutions techniques pour répondre aux besoins de confort (gestion d'énergie, optimisation de l'éclairage et du chauffage), de sécurité (alarme) et de communication (commandes à distance, signaux visuels ou sonores, etc.) que l'on peut retrouver dans les maisons, les hôtels, les lieux publics, etc.[2].

1.3.2 Les domaines de la domotique

La domotique possède plusieurs applications tel :

- le confort : le fait d'automatiser une maison apporte un confort, tel que la possibilité d'ouvrir une porte sans bouger, ou un portail, changer la température de la maison selon l'environnement.
- Les économies d'énergie : être capable de gérer sa consommation d'énergie, par exemple oublier d'éteindre une lampe, etc.
- La sécurité : se sentir en sécurité est important, surtout la possibilité de fermer une vanne de gaz, ou l'extinction des feux, etc.

1.4 Les Systèmes Embarqués

1.4.1 Qu'est ce qu'un système embarqué ?

Un système embarqué ou « Embedded System » est défini comme un système électronique et informatique autonome, souvent temps réel, spécialisé dans une tâche précise. Le terme désigne aussi bien le matériel informatique que le logiciel utilisé. Ses ressources sont

généralement limitées spatialement (encombrement réduit) et énergétiquement (consommation restreinte) [1].

De nos jours, les systèmes embarqués sont utilisés presque partout, il suffit de regarder autour de soi pour s'en rendre compte. Le Distributeur de Billet Automatique (DAB), une imprimante multifonctions, console de jeux vidéo, etc.

Utilisés dans divers domaines, les systèmes embarqués ont facilité et automatisés plusieurs tâches faites manuellement par les êtres humains, beaucoup de fonctions mécaniques et hydrauliques se dématérialisent au profit de l'électronique et de l'informatique embarquée, un confort considérable auquel il est de plus en plus difficile de renoncer. Les systèmes électroniques sont de plus en plus présents dans la vie courante, les ordinateurs et micro-ordinateurs sont des systèmes électroniques bien connus, cependant l'électronique se trouve maintenant embarquée dans de très nombreux objets usuels : les téléphones, les agendas électroniques, les voitures, les GPS [1].

Ce sont ces systèmes électroniques enfouis dans les objets usuels qui sont appelés systèmes embarqués, ces derniers font intervenir des technologies différentes, en plus de comporter une partie logicielle et une partie matérielle.



FIGURE 1.2: Some domains of embedded systems[9].

1.4.2 Les caractéristiques d'un système embarqué [7]

Les systèmes embarqués possèdent les caractéristiques suivantes :

- Ce sont des systèmes électroniques et informatiques autonomes qui ne possèdent pas des entrées-sorties standards.
- Le système matériel et l'application sont intimement liés et noyés dans le matériel et ne sont discernables comme dans un environnement de travail classique de type PC.

- La conception de ces systèmes est fiable (avions, système de freinage ABS) à cause de leur utilisations dans des domaines à fortes contraintes mais également parce que l'accès au logiciel est souvent difficile une fois le système fabriqué.
- Les microprocesseurs utilisés s'étendent depuis de simples microcontrôleurs 8bits aux 64bit les plus performants et les plus sophistiqués.
- Le logiciel système inclus s'étend d'un petit programme à un grand logiciel d'exploitation en temps réel (RTOS) avec une interface utilisateur graphique (GUI). Typiquement, le logiciel système inclus doit répondre aux événements d'une manière déterministe et devrait toujours être opérationnel.

1.4.3 Les avantages d'un système embarqué [7]

- ⊖ Utilise généralement un microprocesseur combiné avec d'autres matériels et logiciel pour résoudre un problème de calcul spécifique.
- ⊖ Système électronique et informatique autonome ne possédant pas des entrées/sorties standards.
- ⊖ Le système matériel et l'application sont intimement liés et noyés dans le matériel et ne sont discernables comme dans un environnement de travail classique de type PC.
- ⊖ N'est pas visible en tant que tel, mais est intégré dans un équipement doté d'une autre fonction ; ou dit aussi que le système est enfoui, ce qui traduit plus fidèlement le terme anglais « Embedded ».
- ⊖ La conception de ces systèmes est fiable (avions, système de freinage ABS).
- ⊖ Les microprocesseurs utilisés s'étendent de simples microcontrôleurs 8bits aux 64bit les plus performants et les plus sophistiqués.

1.5 Présentation des Composants

1.5.1 Arduino Uno

La carte Arduino est un micro-contrôleur, c'est à dire une sorte de mini ordinateur qui sert d'interface entre l'environnement (actions, mesures de grandeurs...) et un utilisateur. Elle se programme nativement dans un langage dérivé du C : le langage « Arduino », Ces cartes permettent un accès simple et peu coûteux à l'informatique embarquée.

1.5.2 Qu'est ce qu'un micro-contrôleur ?

Un micro-contrôleur est un circuit intégré (ou IC, Integrated Circuit), qui rassemble sur une puce plusieurs éléments complexes dans un espace réduit. il est doté :

- D'une unité centrale (CPU) ;
- D'une Mémoire de travail ;
- D'une Mémoire de données ;
- D'une Horloge interne ;
- De ports d'entrée/sortie.

Massimo Banzi et David Cuartielles ont créé l'arduino, comme étant un outil de développement pour des projets de design interactif. Ils ont mis œuvre une petite carte électronique programmable et un logiciel multiplateforme, qui sont accessibles à tous dans le but de créer facilement des systèmes électroniques.

Atmega328p

Le Micro-contrôleur **Atmega328p** est un micro-contrôleur ATMEL de la famille AVR 8bits, et se place comme principal composant de l'Arduino, il possède une puissance de calcul satisfaisante dans des environnement assez complexes. Les caractéristiques du Atmega328p [12] :

- Mémoire flash : 32 kB
- Mémoire SRAM : 2 kB
- Mémoire EEPROM : 1 kB
- Cadencement : 16 MHz

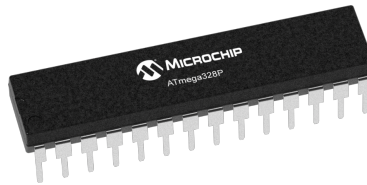


FIGURE 1.3: Atmega328p [12].

Brochage de L'arduino Uno

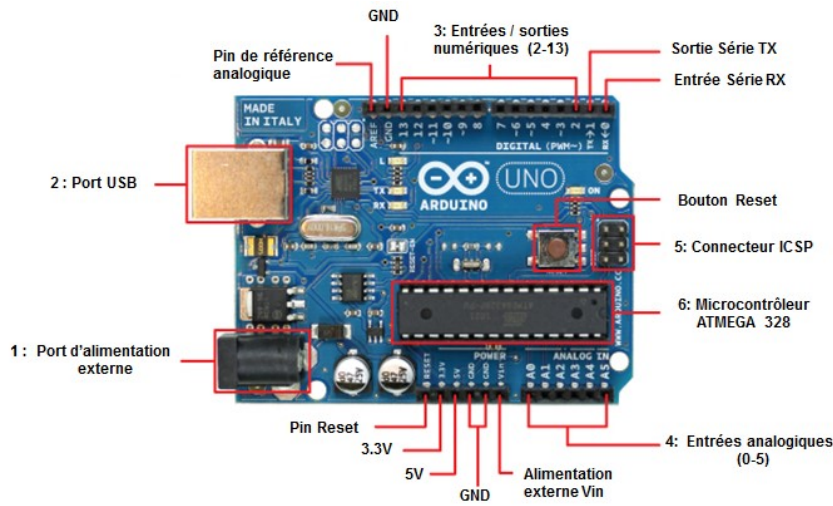


FIGURE 1.4: Pinout of the Arduino Uno [11].

- **1 :** Un port d'alimentation externe pour fonctionner, la carte a besoin d'une alimentation qui est comprise entre 5 et 20 V. On dp conseille en général de l'alimenter plutôt entre 7 et 12V pour garder une marge en basse tension et éviter que le circuit ne chauffe trop (car le régulateur de tension disperse toute surtension n en chaleur), cette tension doit être continue et peut par exemple être fournie par une pile 9V. Un régulateur se charge ensuite de réduire la tension à 5V pour le bon fonctionnement de la carte.
- **2 :** Un port USB Permet de communiquer avec la carte et de l'alimenter en 5V.
- **3 :** Les entrées/sorties numériques : 14 entrées/sorties numériques (Digital) dont 6 peuvent assurer une sortie PWM, ils peuvent actionner de nombreux composants (LED, transistor, etc.) mais elles ne peuvent pas fournir beaucoup de courant (40 mA pour une carte Arduino UNO). Pour piloter des circuits de plus forte puissance, il faut passer par des transistors ou des relais.
- **4 :** Les entrées analogiques (6 entrées) lui permettent de mesurer une tension variable (entre 0 et 5 V) qui peut provenir de capteurs ou d'interfaces diverses (potentiomètres, etc.).
- **5 :** Connecteur In-Circuit Serial Programming (ICSP) pour programmer l'arduino grâce à un autre arduino.
- **6 :** Micro-contrôleur ATmega328P.

Le tableau 1.2 présente quelques caractéristiques de l'arduino uno, et la figure 1.4 présente sa structure externe.

Items	Features/Ratings
Microcontroller	ATmega328
Operating Voltage	5V
InputVoltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital /O pins	14 (6 provide PWM output)
Analog Input pins	6
DC current per I/O pin	40mA
DC current for 3.3V pin	50mA
Clock Speed	16MHz
Flash Memory	32Kbits (0.5 Kbits used by bootloader)
SRAM	2 Kbits

TABLE 1.2: Arduino Uno characteristics [3]

1.5.3 Pourquoi arduino ? [8]

- **Le prix** : les cartes Arduino sont relativement peu coûteuses comparativement aux autres plates-formes. La moins chère des versions du module Arduino peut être assemblée à la main, (les cartes Arduino pré-assemblées coûtent moins de 5000 Dinars).
- **Multi Plateformes** : le logiciel Arduino, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux. La plupart des systèmes à micro-contrôleurs sont limités à Windows.
- **Environnement de développement simple et claire** : l'environnement de Programmation Arduino (le logiciel Arduino IDE) est facile à utiliser pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.
- **Logiciel open source** : On peut l'utiliser et le modifier librement.
- **Matériel open source** : Les plans des cartes Arduino sont publiés sous une licence Creative Commons, de sorte que les concepteurs de circuits expérimentés peuvent créer leur propre version du module, l'étendre et l'améliorer. Même les utilisateurs relativement inexpérimentés peuvent créer la version maquette du module afin de comprendre son fonctionnement et d'économiser de l'argent.

1.5.4 Câbles jumper et résistances

Câbles jumper

Les fils ou câbles jumper sont des fils de liaison qui peuvent être de prises mâles-femelles ou mâles-mâles ou femelle-femelle, la figure 1.5 nous montre à quoi ressemble ces fils.



FIGURE 1.5: Jumper Cables[13].

Résistance

Une résistance est un composant électronique ou électrique dont la principale caractéristique est d'opposer une plus ou moins grande résistance (mesurée en ohms) à la circulation du courant électrique.



FIGURE 1.6: Resistances [24].

1.5.5 Module GSM SIM800L

Un modem GSM est un modem sans fil qui fonctionne avec un réseau sans fil GSM. GSM signifie Global System for Mobile communications, cette architecture est utilisée pour la communication mobile.

Le module SIM800L GSM/GPRS est un modem GSM miniature, et en plus de sa taille miniature, il consomme moins de 2A, il est utilisé pour accomplir presque tout ce qu'un téléphone portable normal peut faire ; envoyer un SMS, passer ou recevoir des appels téléphoniques, se connecter à Internet via GPRS, TCP/IP, etc. Il fonctionne en utilisant des commandes AT².



FIGURE 1.7: SIM800L GSM Module[14].

Son brochage est simple, et la figure 1.8 présente les différents pins qui caractérise ce module. Les pins qui seront utilisés dans ce mémoire sont les suivants :

- VCC : Entrée courant principale 5V.
- GND : Ground.
- RX : Receiving pin ou pin de réception.
- TX : Transmitting pin ou pin de transmission.

2. AT est une abréviation de "ATtention" qui sont des commandes spécifiques aux modems.

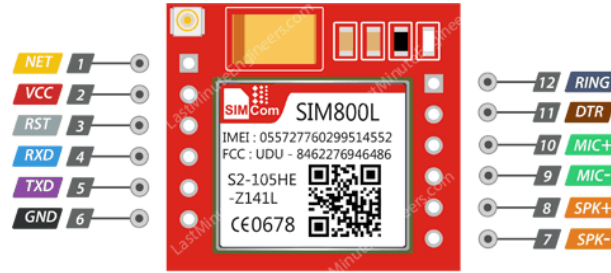


FIGURE 1.8: Pinout of GSM SIM800L Module[14].

1.5.6 Circuit de Mesure : Capteurs de Gas

Qu'est ce qu'un Capteur de Gas ?

Un capteur de gas est défini comme un composant dont au moins une de ses propriétés physiques change quand il est soumis à un changement d'environnement gazeux. D'une manière générale, un capteur est composé de deux éléments principaux : l'élément sensible et le transducteur. L'élément sensible est le cœur du capteur, sur lequel se passe la réaction avec l'espèce gazeuse. Le transducteur est le dispositif permettant la conversion du résultat de la réaction entre le gas et l'élément sensible en un signal facilement mesurable (signal électrique ou optique).

Performances d'un capteur de gas

— Sensibilité

La sensibilité, par définition, est le rapport de la variation de la réponse électrique du capteur à celle de la concentration du gas :

Cependant, dans le domaine des capteurs MOX, on utilise parfois d'autres paramètres pour traduire la sensibilité du capteur. R_s/R_o sensibilité relative. $(R_s - R_o)/R_o$ sensibilité fractionnelle. $\ln(R_s/R_o)$ sensibilité logarithmique. Les indices s et o correspondent respectivement à la mesure sous gas et sous air quelle que soit la définition utilisée. Plus la valeur du paramètre est grande, mieux le capteur réagit au gas. Actuellement, les capteurs de gas sont suffisamment sensibles.

— Stabilité

La stabilité est définie comme étant la capacité d'un capteur à répondre à un certain gas en présence des gas interférents. C'est le paramètre le plus important car le capteur est souvent utilisé pour détecter un gas dans une atmosphère contenant plusieurs gas. Actuellement, les capteurs à base d'oxydes métalliques souffrent d'un manque important de sélectivité et de nombreuses méthodes sont étudiées pour résoudre ce problème.

— Température de fonctionnement

Le processus d'adsorption et de désorption des molécules de gas à la surface du capteur dépend fortement de sa température de fonctionnement. Afin de garantir une bonne réversibilité des phénomènes d'adsorption et de désorption, les capteurs MOX fonctionnent à température élevée, typiquement entre 300°C et 500°C. La réduction de la température de fonctionnement entraîne une diminution de la puissance électrique consommée. Dans ce but, des recherches sur des capteurs fonctionnant à des températures plus basses ou à température ambiante sont en cours.

— Temps de réponse

Il exprime le temps nécessaire que met la valeur de sortie du capteur pour se stabiliser lorsque les conditions de mesure varient brutalement d'un état à un autre. Le temps de réponse est pris entre 10% et 90% de la valeur stabilisée. Dans le cas des capteurs de gas, cette valeur dépend essentiellement de la cinétique des réactions chimiques mises en jeu. Il est à noter qu'il est souvent très difficile d'estimer ce temps sans prendre en compte celui du banc de caractérisation qui peut être soit plus petit, soit équivalent, soit bien plus important suivant les conditions de mesure.

La série des modules MQ-x

Les modules MQ-x sont des appareils de mesure de la densité d'un type de gas présents dans l'atmosphère, que ce soit un gas combustible (CH_4 , Liquefied petroleum gas (LPG C_3H_8), etc.) ou toxique (CO , CO_2 , etc.). Ce type d'appareil est utilisé le plus souvent dans des environnements industriels ; cependant des tentatives existent pour le faire entrer dans les usages domestiques (détecteur de fuites de méthane).

Les différents types des capteurs MQ

— Le capteur MQ-2

Le Capteur de gaz MQ2 est utile pour la détection des fuites de gaz (dans la maison et l'industrie). Il peut détecter le H_2 , le GPL, le CH_4 , le CO , l'alcool, la fumée, le propane. Basé sur son temps de réponse rapide, les mesures peuvent être prises dès que possible, de plus, la sensibilité peut être réglée par le potentiomètre.



FIGURE 1.9: MQ-2 Module[21].

— Le capteur MQ-5

Le capteur de gaz MQ-5 a une grande sensibilité au méthane, au propane et au butane, et pourrait être utilisé pour détecter le méthane et le propane. Le capteur pourrait être utilisé pour détecter différents gaz combustibles, en particulier le méthane, il est à faible coût et convient à différentes applications.



FIGURE 1.10: MQ-5 Module[20].

— Le capteur MQ-7

Le capteur de gas monoxyde de carbone MQ-7 est un capteur de gaz semi-conducteur qui détecte la présence de gaz monoxyde de carbone à des concentrations de 10 ppm à 10000 ppm. L'interface de tension analogique simple du capteur nécessite une seule broche d'entrée analogique à partir de votre micro-contrôleur. Le capteur de gas monoxyde de carbone MQ-7 détecte la concentration de gas monoxyde de carbone dans l'air et évite sa lecture en tant que tension analogique. Le capteur peut fonctionner à des températures comprises entre -10 et 50°C et consomme moins de 150 mA à 5 V.



FIGURE 1.11: MQ-7 Module[22].

— Le capteur MQ-9

Il est utilisé dans certains équipements de détection de gaz pour le monoxyde de carbone (CO), le méthane (CH₄) et le gaz de pétrole liquéfié (GPL). Le capteur peut être utilisé pour détecter différents gaz, du CO et des gaz combustibles, contenus dans l'atmosphère.



FIGURE 1.12: MQ-9 Module[23].

Brochage du Module MQ-2

Le module utilisé dans ce mémoire sera le module MQ-2, le plus populaire parmi la série MQ-x, et son brochage, montré sur la figure 1.13 nous montre les différents pins ce module :

- VCC : Entrée courant (5V).
- : GND : Ground.
- A0 : Analog 0, ce pin va permettre d'envoyer le taux de mesure vers l'arduino.
- D0 : ce pin permet de transmettre la classe, 1 pour l'oxygène, et un 0 pour les autres gas.

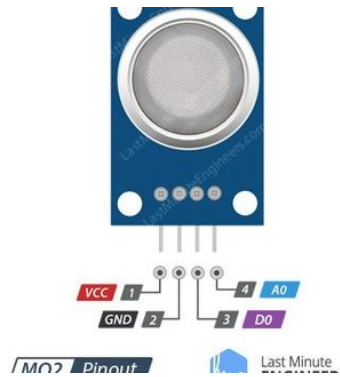


FIGURE 1.13: Pinout of MQ-2 Module[10].

PPM (partie par million) ?

Partie Par Million (PPM) est un terme scientifique, c'est une mesure du taux de gas dans l'environnement ; Il s'agit du ratio d'un gas par rapport à un autre.

Par exemple : pour 1000 ppm de CO, et pour 1 million de molécules de gas, on comptera 1,000 molécules de CO et 999 000 molécules d'autres types de gas. Le graphe nous montre la concentration d'un gaz en partie par million (ppm) en fonction du rapport de résistance du capteur (R_S/R_0) [10].

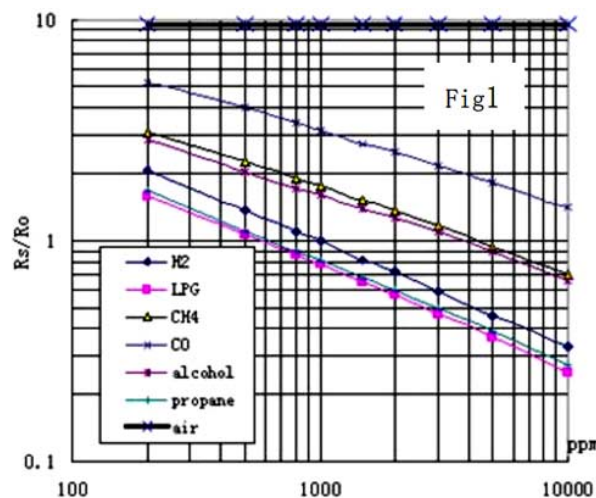


FIGURE 1.14: Sensivity's Characteristics of MQ-2 Module[10].

Le graphe, indique la concentration d'un gaz en partie par million (ppm) en fonction du rapport de résistance du capteur (R_S/R_0).

R_S est la résistance du capteur qui change en fonction de la concentration de gas. R_0 est la résistance du capteur à une concentration connue sans la présence d'autres gaz, ou à l'air frais. Pour l'air, $\frac{R_S}{R_0} = 9,8$ pour le capteur de gaz MQ2.

Remarque 1.5.1 *D'après le graphe, nous pouvons voir que la concentration minimale que nous pouvons mesurer est de 100 ppm et que la concentration maximale est de 10000 ppm, en d'autres termes, nous pouvons obtenir une concentration de gaz comprise entre 0,01 % et 1 %.*

Composants et Fonctionnement du Module MQ-2

Dans ce mémoire, nous avons utilisé le module MQ-2 qui est souvent utilisé pour la mesure du CO dans l'air. Nous allons dans cette section introduire les composants du module et comment ce dernier fonctionne-t-il.

Le capteur est enfermé dans deux couches de mailles fines en acier inoxydable appelées réseau anti-explosion(anti-explosion network). Cela garantit que l'élément chauffant à l'intérieur du capteur ne provoquera pas d'explosion, car nous détectons des gaz inflammables [10].

Ces mailles assurent également la protection du capteur et filtre les particules en suspension afin que seuls les éléments gazeux puissent passer à l'intérieur du module [10].



FIGURE 1.15: MQ2 Gas Sensor Parts[10].

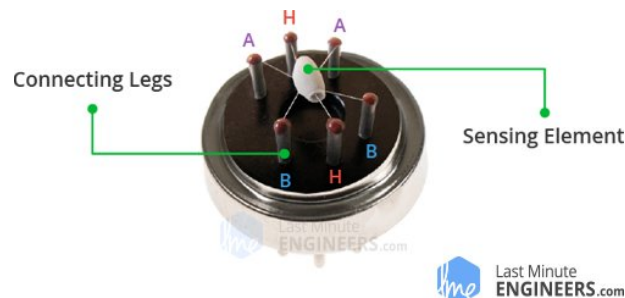


FIGURE 1.16: Inside Gas Sensor Internal Structure of MQ-2 Sensor[10].

Voici à quoi ressemble le capteur lorsque la protection est retirée. La structure en forme d'étoile est formée par l'élément de détection et six pattes de connexion qui s'étendent au-

delà de la base en bakélite³. Sur six, deux fils (H) sont chargés de chauffer l'élément de détection et sont connectés via une bobine Nichrome⁴, un alliage⁵ conducteur bien connu [10].

Les quatre fils restants (A et B) responsables des signaux de sortie sont connectés à l'aide de fils de platine. Ces fils sont connectés au corps de l'élément de détection et transmettent de petits changements dans le courant qui traverse l'élément de détection [10].

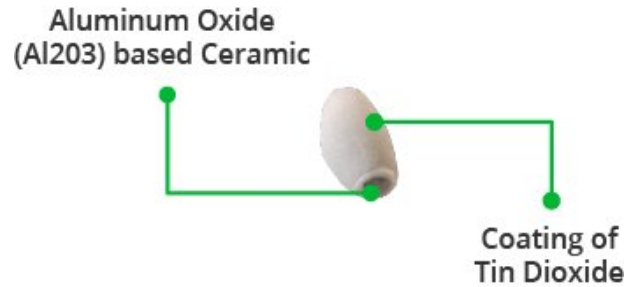


FIGURE 1.17: Sensing-Element[10].

L'élément chauffant est composé de céramique à base d'oxyde d'aluminium (Al_2O_3) et possède un revêtement de dioxyde d'étain (SnO_2). Le dioxyde d'étain est le matériau sensible aux gas combustibles. Cependant, le substrat en céramique augmente simplement l'efficacité de chauffage et garantit que la zone du capteur est constamment chauffée à une température de fonctionnement⁶ [10],

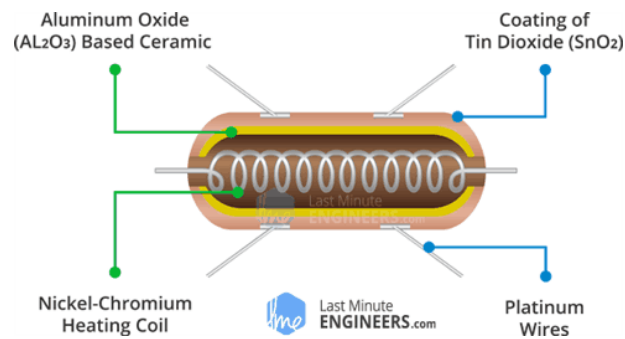


FIGURE 1.18: Internal structure of sensing element[10].

Ainsi, la bobine de nichrome et la céramique à base d'oxyde d'aluminium forment un système de chauffage ; tandis que les fils de platine et le revêtement de dioxyde d'étain

3. le Bakélite est un plastique synthétique crée par Leo Baekeland en 1909.

4. Le Nichrome est une sorte d'alliage non magnétique capable de résister à de hautes températures.

5. Un alliage est une fusion de plusieurs éléments métalliques pour donner un autre éléments métallique ou chimique si le métaux est fusionné à un élément chimique.

6. C'est d'ailleurs pour lequel le module doit être chauffée au préalable avant de commencer à mesurer.

forment un système de détection [10].

Comment fonctionne un capteur de gas ?

Lorsque le dioxyde d'étain est chauffé dans l'air à haute température, l'oxygène est attiré vers la surface. Dans l'air pur, les électrons donneurs du dioxyde d'étain sont attirés vers l'oxygène qui est absorbé à la surface du matériau de détection. Cela empêche la circulation du courant électrique. En présence de gas réducteurs, la densité de l'oxygène absorbé diminue au fur et à mesure qu'il réagit avec les gas réducteurs. Des électrons sont ensuite libérés dans le dioxyde d'étain, permettant au courant de circuler librement à travers le capteur [10].

1.5.7 Le Relai

Le relai est constitué d'un électroaimant et d'un contacteur mécanique. Lorsque le courant est suffisamment important sur la borne d'entrée, le solénoïde se magnétise ce qui force le contacteur à fermer le circuit de puissance.

Le relai est un élément important lorsqu'il s'agit de travailler sur des courants plus élevés par rapport à d'autres, Par exemple : l'arduino fait passer seulement 5V, si on essaye d'alimenter une lampe qui nécessite 12v, l'arduino n'est pas suffisant et si on essaye d'alimenter l'arduino avec 12v, celui-ci sera endommagé, c'est là qu'intervient le relai, il permet de créer une communication entre l'arduino et la lampe.

Le module du relai possède trois pins de chaque côté, Courant Positif (+), Courant Negatif (-) et Le signal(S) d'un côté, Normally Closed(NC), Normally Open (NO), et Common(COM). Les pins NC, NO et COM sont définis comme suit :

- NC : Normally Closed veut dire que le contact est effectué et que le courant passe ; si le relai reçoit un signal celui-ci va être ouvert et le courant coupé.
- NO : l'opposé du NC, le courant est coupé, si un signal est reçu, le courant va passer et le pins sera donc fermé.
- COM : ce pin va faire passer le courant de 12V, et en dépendance avec les pins NC et NO.

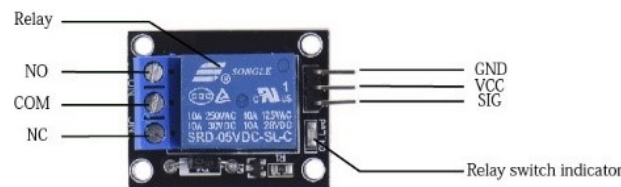


FIGURE 1.19: Relay[19].

1.5.8 Led, bi-led et buzzer

Il existe de nombreux choix pour communiquer des informations entre un produit et un utilisateur, l'un des choix les plus courants pour la communication audio est un buzzer. Le buzzer est un haut-parleur de faible puissance qui va émettre un son en fonction de la fréquence et l'amplitude de vibration, celui-ci est utilisé dans la plupart des applications et projet afin de signaler une erreur, une notification, etc.



FIGURE 1.20: Buzzer [17].

La light-emitting diode (LED) est une petite source de lumière qui ne fait passer le courant que d'un seul sens, elle joue aussi le rôle de diode, d'où le nom diode.



FIGURE 1.21: Light-Emmiting Diode (LED) [16].

Afin de minimiser le nombre de composants et l'encombrement, il existe un autre type de led : une LED bicolore est capable d'émettre deux couleurs de lumière différentes, généralement rouge et verte, plutôt qu'une seule couleur.

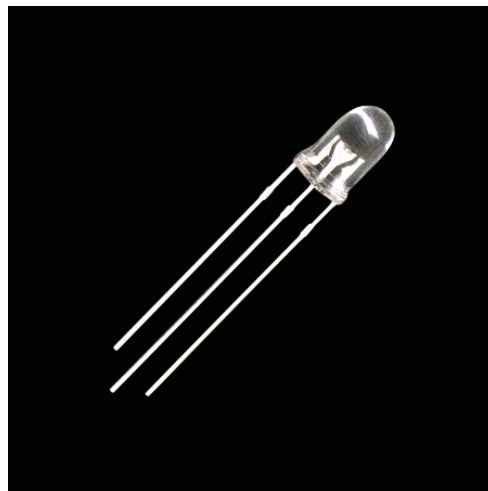


FIGURE 1.22: Bi-color Led [15].

1.5.9 LCD I2C Crystal

LCD ou Liquid Crystal Display est un module d'affichage qui permet d'afficher 32 caractères ASCII dans 2 lignes de 16 caractères chacune, les caractères seront d'une forme de petit carrées (pixelisés).



FIGURE 1.23: LCD I2C Crystal Display

Le module d'affichage est optionnel, mais parfois, il est utile de vérifier le niveau d'un gas en utilisant ce dernier comme affichage, donc le LCD I2C est l'outil convenable pour cette tâche.

1.5.10 Prototype Shield V5.0 et BreadBoard

Le shield de Prototypage est un outil qui permet de simplifier le développement d'un mini projet et possède plusieurs sorties 5v et Gnd(Ground) avec deux leds et un bouton qui peuvent être utilisées afin de réduire l'utilisation des pins 5v et gnd limitées de l'arduino. Cet outil vient avec une breadboard qui peut être intégrée au shield.

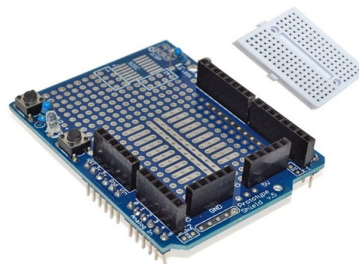


FIGURE 1.24: Protoyp Shield and BreadBoard

1.6 Présentation de l'IDE Arduino

L'IDE⁷ Arduino possède un éditeur de texte, une console, une barre de notifications, des boutons et plusieurs menus. Cet IDE va permettre de compiler et téléverser par l'intermédiaire de avrdude⁸, les codes sources dans l'arduino et communiquer avec ce dernier. La figure 1.25 nous montre à quoi ressemble cette IDE.

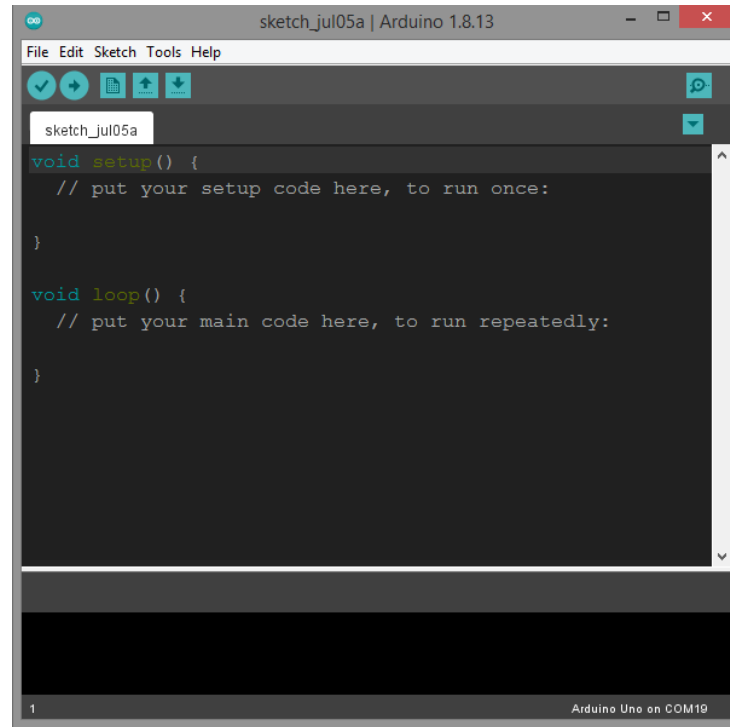


FIGURE 1.25: Arduino IDE

Afin de téléverser le code source, il est nécessaire de le compiler(Vérifier), et ceci par le biais du bouton



FIGURE 1.26: Verify Button

et pour le téléverser, le bouton



FIGURE 1.27: Upload Button

7. Integrated Development Environment.

8. avrdude est un programme de ligne de commande, il s'agit du compilateur par défaut de l'arduino.

Avant de téléverser, il est important de déterminer le modèle de la carte et le port de téléversement ; On sélectionne le modèle (Tools > Board) et le port (Tools > Port⁹), ensuite on effectue le téléversement.

Il est parfois utile d'utiliser le moniteur serial afin de communiquer avec l'arduino, c'est pour cela que le bouton



FIGURE 1.28: Serial Monitor Button

peut être utilisé, ainsi que dans le cas où l'on souhaite communiquer avec un appareil qui accepte la communication serial.

Remarque 1.6.1 *Il existe aussi un autre type de moniteur : le moniteur plotter, qui est capable de dessiner des courbes.*

1.7 Principe de la logique floue

Le concept de logique floue a été proposé dans les années 60 par Lotfi Zadeh, un mathématicien et informaticien iranien, pour répondre aux limites de la logique classique. Prenons un exemple très simple sur la vitesse à laquelle vous roulez en voiture sur l'autoroute. Si on raisonnait en logique classique, on pourrait dire par exemple : « Si je roule à moins de 100 km/h, je roule lentement et si je roule à plus de 100 km/h, je roule vite ».

C'est un raisonnement un peu simpliste, ça voudrait dire qu'à partir du moment où vous dépassez les 100 km/h, tout d'un coup, vous vous mettez à rouler vite. Encore une fois, ce n'est pas de cette façon que réfléchit le cerveau humain, nous fonctionnons beaucoup plus par nuances, par transition, et c'est ce que la logique floue permet d'apporter. Dans ce dernier exemple, on dirait plutôt qu'on roule lentement sur l'autoroute en dessous de 70 km/h, qu'on roule vite au-dessus de 130 km/h, et qu'entre 70 et 130 km/h, on n'est ni lent, ni rapide, on est un peu des deux en quelque sorte [25].

Dans l'exemple précédent, on a fait intervenir une seule variable (la vitesse sur l'autoroute dans ce cas). Bien évidemment, il n'y a pas de limites sur le nombre de variables à utiliser pour prendre une décision, c'est d'ailleurs rare que l'on fasse un choix ou qu'on prenne une décision à partir d'une seule variable [25].

9. Il faut s'assurer d'avoir branché la carte et de sélectionner le bon port, si celui-ci s'avère être un mauvais port, une erreur s'affichera sur la console.

Quelques concepts de la logique floue

Avant de rentrer plus dans le détail de la logique floue, on se rend compte que l'exemple précédent nous a permis d'aborder quelques concepts clés de la logique floue :

- En logique floue, une valeur peut appartenir à plusieurs ensembles à la fois contrairement à la logique classique. Par exemple, en reprenant notre exemple de vitesse sur l'autoroute, 90 km/h en logique classique, c'est une vitesse lente ; alors que 90km/h en logique floue, ce n'est pas totalement rapide mais ce n'est pas totalement lent non plus.
- En logique floue (mais aussi en logique classique), on met en place un ensemble de règles (qu'on appellera règles d'inférence floue par la suite) de la forme « Si. . . , alors. . . ». Dans notre exemple, si la vitesse est lente ET il n'y a aucun obstacle alors on augmente la vitesse. La fonction « ET » en raisonnement logique s'interprète comme une intersection : pour que je diminue la vitesse, il faut qu'à la fois on roule vite et il y'a un obstacle devant nous. D'autres fonctions existent, notamment la fonction « ou » qui, en raisonnement logique, s'interprète comme une union : Si au moins un des deux critères est respecté (et pas nécessairement les deux comme avec une intersection), alors. . . Lorsque l'on utilise des variables au sein de nos règles, il faut être capable de donner une valeur ou de noter ces variables. Dans le cas de la vitesse sur l'autoroute, ça se comprend facilement puisqu'une vitesse peut se mesurer en km/h[25].

Les systèmes d'inférence floue

On parle plutôt de systèmes d' « inférence¹⁰ floue » plutôt que de système de « logique floue », un système d'inférence floue est un système composé de trois éléments : la fuzzification¹¹, le moteur d'inférence et la défuzzification.

- **Fuzzification** : "Fuzzification may be defined as the process of transforming a crisp set to a fuzzy set or a fuzzy set to fuzzier set. Basically, this operation translates accurate crisp input values into linguistic variables" [26]. La fuzzification est un processus qui permet de transformer des valeurs binaires(précise) en valeurs réelles(floues).
- **Moteur d'inférence** : C'est un composant du système qui applique des règles logiques à une base de connaissances pour en déduire de nouvelles informations. Les premiers moteurs d'inférence étaient des composants de systèmes experts. Le système d'expert typique consistait en une base de connaissances et un moteur d'inférence[26].

10. Inférer : conclure, induire, etc.

11. Fuzzy logic en anglais d'où le terme fuzzification et defuzzification.

- **Defuzzification** : L'inverse de l'opération de fuzzification, elle peut être défini comme le processus de transformation d'un ensemble flou en un ensemble net ou de conversion d'un membre flou en un élément net.

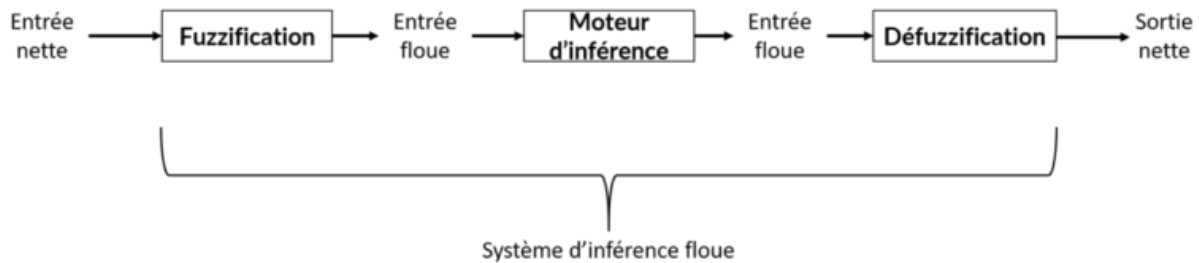


FIGURE 1.29: Un système d'inférence floue

1.8 Comparaison avec des travaux antérieurs

Nous avons fait une étude comparative avec un travail [4] qui a déjà été fait auparavant et nous avons déduit les points suivant :

Notre Système	Autre système [4]
Max Voltage utilisé : 5V	Max Voltage utilisé : 12V
Module GSM utilisé : Sim800L	Module GSM utilisé : Sim800
Consommation électrique : 2Wh ¹² Sim800L	Consommation électrique : 2.25Wh
Pas de driver Motor	Driver Motor L298N
Relai Utilisé	Relai non utilisé

TABLE 1.3: Tableau des comparaisons

On constatera, d'après le tableau 1.8 que de net améliorations concernant la consommation électrique ont été présentés avec l'utilisation de moins de module et un encombrement réduit ¹³.

1.9 Conclusion

Nous avons vu dans cette partie que le monoxyde de carbone est l'une des principales causes de décès et nous avons présenté quelques informations sur le taux de CO qui pourrait être mortel dans la première section. Dans la seconde section, nous avons présente ce que sont les systèmes embarqués et la domotique ainsi que leurs domaines d'utilisation. Et

13. La dimension n'est pas présente dans ce travail mais diffère nettement.

pour conclure, nous avons donné les principaux composants qui ont fait ce système, nous avons expliqué le fonctionnement du détecteur de gaz, ainsi que le concept de logique floue et le système d'inférence en donnant des exemples.

CHAPITRE 2

ASSEMBLAGE : CONCEPTION, RÉALISATION ET BRANCHEMENT

2.1 Introduction

Dans cette partie pratique, nous illustrerons le fonctionnement du travail avec deux diagrammes, nous expliquerons l'application de la logique floue dans notre système et nous détaillerons le branchement nécessaire à effectuer. Enfin, nous exposerons quelques parties du code utilisé.

2.2 Schéma principal et diagramme d'activité

Nous allons commencer par introduire les deux diagrammes suivies dans ce mémoire, un principal et l'autre consistera en un diagramme d'activité.

2.2.1 Schéma principal

Dans le schéma principal¹(Figure 2.1), tout se produit à domicile (utilisation domestique). L'arduino est en liaison direct avec le détecteur de gas et obtient le taux de présence du gaz dans l'air de ce dernier. Si ce taux est assez élevé, une alarme sera déclenché ; si ce taux continue de s'élever, le système d'aération devra être activé afin de minimiser le risque ; si celui-ci ne réussit pas à réduire le risque et le niveau de gas continue de s'élever alors un appel devra être effectué vers 3 personnes consécutivement.

1. Schéma crée avec "draw.io".

Alarming Gas Sensor

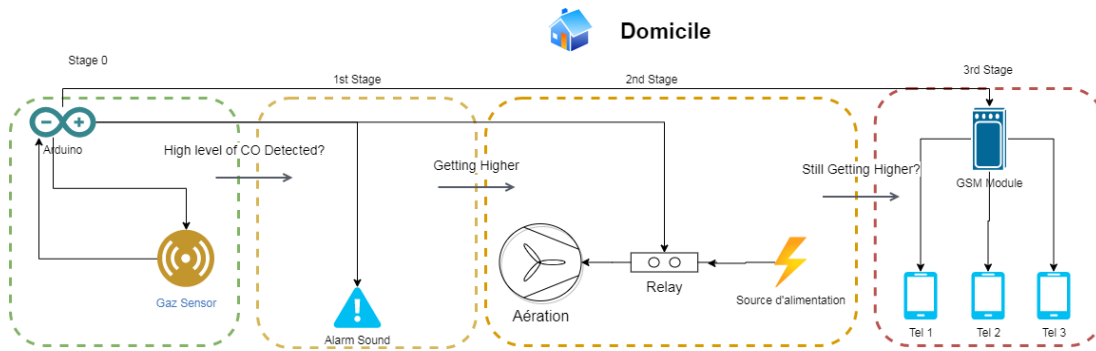


FIGURE 2.1: Principal Diagram

2.2.2 Diagramme d'activité

Dans le diagramme d'activité (Figure 2.2), l'arduino agit comme principal nœud qui va effectuer l'action² d'obtenir des valeurs du sensor MQ-x ; cette action va être suivie de condition³, qui va nous donner comme conséquence des actions qui seront effectuées en suivant ces conditions. On précise dans le diagramme que chaque action est effectuée en fonction de la condition obtenue, plus la valeur A0 est élevée plus le nombre d'actions est élevé ; donc les 3 conditions sont vraies et les 3 actions sont produites en même temps⁴.

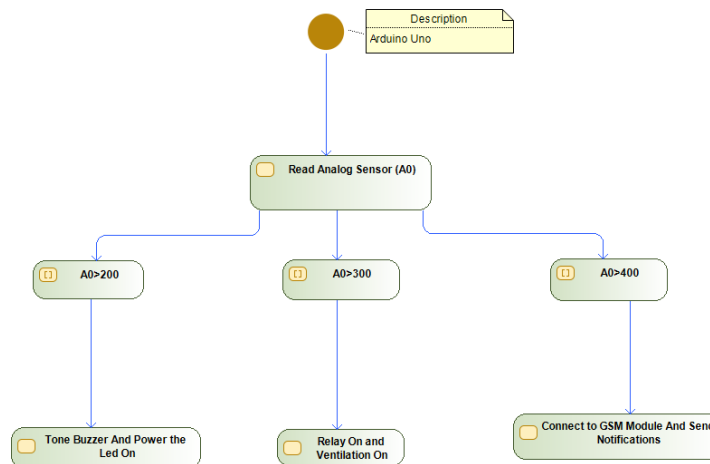
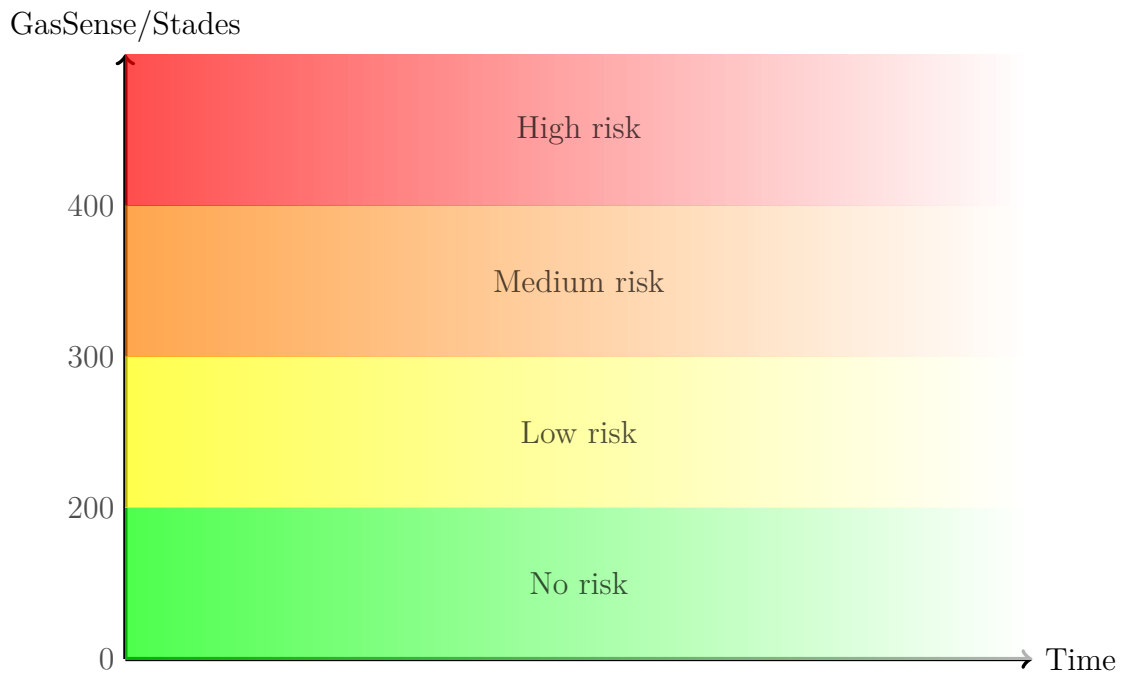


FIGURE 2.2: Activity Diagram

2. On parle d'action (ou Opaque Action) lorsqu'il s'agit d'un diagramme d'activité.
3. On remarque la différence entre action et condition (Conditional Node) sur les crochets de la condition.
4. L'inconvénient ici c'est que les actions sont produites consécutivement et non en parallèles, et ceci est un inconvénient de la carte de développement Arduino.

2.2.3 Niveaux d'alarmes

Lorsque l'on parle de niveaux ou stades, on définit une logique conditionnelle ou l'alarme devrait se déclencher ; Le graphe de la figure nous indique la valeur d'entrée du module MQ-2, et le niveau dans lequel un stade sera exécuté :



— Pour le stade 0 : Aucun risque et aucune alarme.

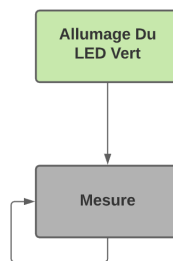


FIGURE 2.3: Stade 0

- Pour le stade 1 : un risque minimum, donc un buzzer et un led s’activeront pour une notification mineure.

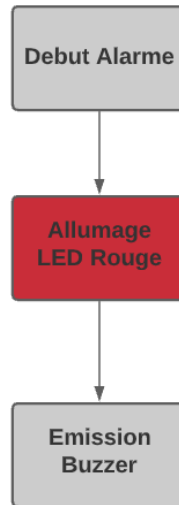


FIGURE 2.4: Stade 1

- Pour le stade 2 : un risque moyen, un relai devra impérativement activer le système d’aération.

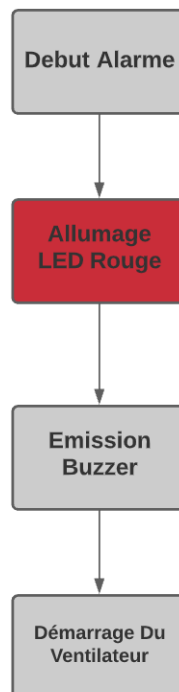


FIGURE 2.5: Stade 2

- Pour le stade 3 : un risque élevé, le module GSM devra effectuer un appel vers 3 personnes consécutivement.

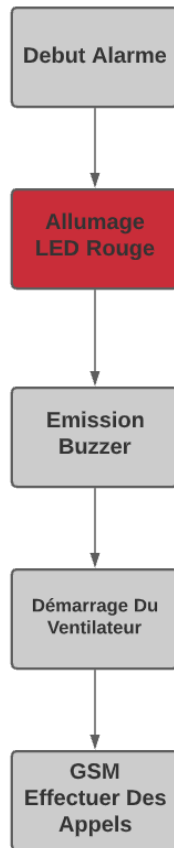


FIGURE 2.6: Stade 3

2.2.4 Application de la logique floue

Nous avons vu dans la partie présentation que la logique floue utilisait un système d'inférence, c'est pour cela que nous allons maintenant définir de quoi est constitué ce système.

Identification des entrées/sorties

Dans ce travail, une seule variable d'entrée et une seule variable de sortie ont été prises en compte. Pour la variable d'entrée, ce sera le taux de voltage converti ⁵ qui prend trois fonctions d'appartenance qui sont faible, moyen, et haut dans un univers de discours bien déterminé allant de 0 à 600. La variable d'entrée est reliée au pin A0. Pour la variable de sortie, ce sera le stade d'alarme qui devra s'activer (stade 1, 2 ou 3) et qui va prendre ses valeurs entre 0 et 3.

5. la fonction `analogRead(pin)` permet de convertir le voltage en entrée du pin spécifié en valeur entre 0 et 1023.

Fonction d'appartenance de la variable d'entrée/sortie

La variable d'entrée est représentée par un ensemble de trois fonctions d'appartenance {High, Low, Medium} (voir Figure 2.7) :

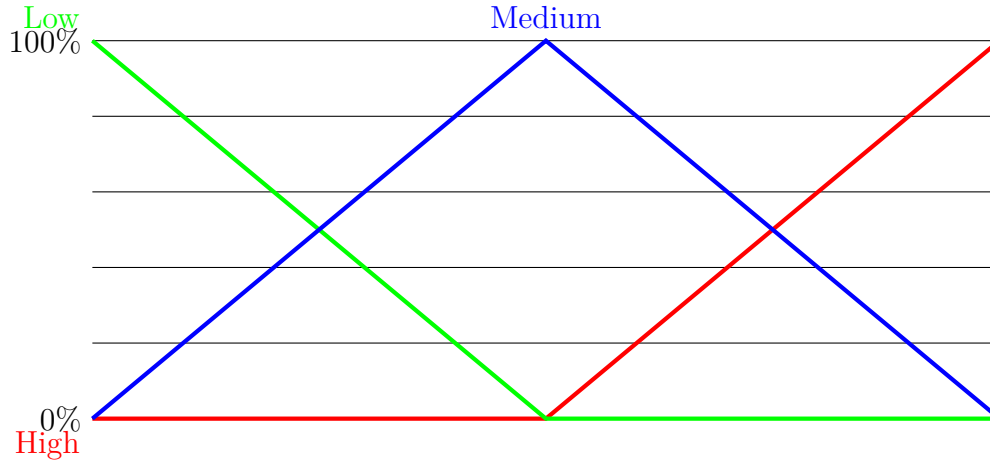


FIGURE 2.7: Fonctions d'appartenance de la variable d'entrée.

La variable de sortie est représentée aussi par un ensemble de trois fonctions d'appartenance {High, Low, Medium} (voir Figure 2.8) :

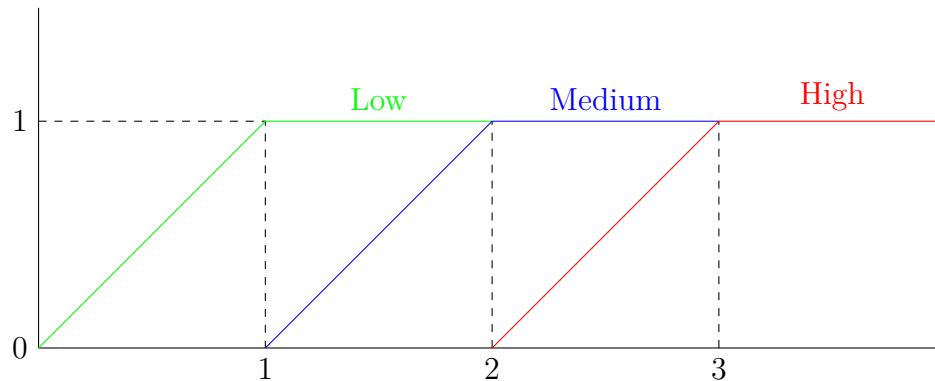


FIGURE 2.8: Fonctions d'appartenance de la variable de sortie.

Règles du système d'inférence

Afin de comprendre comment le système fonctionne, nous allons citer les règles de bases utilisées dans ce système :

- Si le taux de voltage est faible à plus de 50% et moyen⁶ à moins de 50% alors il n'y a aucun risque et aucune alarme.

6. On dit que le taux de voltage appartient à la classe faible et moyen à un certain pourcentage ou degré d'appartenance.

- Si le taux de voltage est faible à moins de 50% et moyen à plus de 50% alors activation du stade d'alarme 1.
- Si le taux de voltage est moyen à plus de 50% et haut à moins de 50% alors activation du stade d'alarme 2.
- Si le taux de voltage est haut à plus de 50% et moyen à moins de 50% alors activation du stade d'alarme 3.

2.3 Quelques parties du Code

2.3.1 Présentation des bibliothèques

SoftwareSerial.h

La bibliothèque Software serial comme présente sur le site arduino : "The SoftwareSerial library has been developed to allow serial communication on other digital pins of the arduino, using software to replicate the functionality (hence the name "SoftwareSerial"). It is possible to have multiple software serial ports with speeds up to 115200 bps. A parameter enables inverted signaling for devices which require that protocol" [18], a été développée pour effectuer une communication serial⁷ entre l'arduino et des appareils qui possèdent des ports RX et TX en utilisant un protocole de communication UART (Universal asynchronous receiver-transmitter).

Wire.h

Cette bibliothèque permet la communication avec les modules I2C⁸, cette bibliothèque est importante, même si elle n'est pas présente dans la plupart des programmes, celle-ci est en liaison avec la bibliothèque Arduino.h, elle permet aussi la communication avec des appareils TWI (Two Wire Interface).

Regexp.h

Une bibliothèque crée par Nick Gammon qui permet de chercher ; de diviser en segment, remplacer et supprimer des lettres dans une chaine de caractères, cette bibliothèque est nécessaire lorsque l'on veut diviser des réponses obtenues du module GSM SIM800L.

Cette bibliothèque implémente plusieurs fonction tel que match(regular expression), getCapture(buffer, int), Target(char*) qui sont des fonctions callback(), appelés de l'objet MatchState.

7. On remarquera que le moniteur serial est utilisée pour communiquer avec l'arduino.

8. I2C est un protocole de communication, qui permet le transfert de données bit par bit (SDA(data line) et SCL(clock line)).

eFLL (Embedded Fuzzy Logic Library)

Cette bibliothèque développée par le Groupe de Recherche Robotique(RRG), est une bibliothèque versatile, légère et efficace pour développer des systèmes qui utilisent la logique floue. La bibliothèque utilise le processus : (MAX-MIN) et (Minimum Mamdani) pour l'inférence et la composition et (CENTRE DE ZONE) pour la défuzzification dans un univers continu [28].

2.3.2 Quelques fonctions et leur définition

void setup()

Une fonction appelée au démarrage de la carte, cette fonction est utilisée pour initialiser les variables, le sens des broches, cette fonction n'est exécutée qu'une seule fois, après chaque mise sous tension ou reset (ré-initialisation) de la carte Arduino.

void loop()

Une fonction qui fait exactement ce que son nom suggère et s'exécute en boucle sans fin, permettant au programme de s'exécuter et de répondre.

void pinMode(Pin, comportement du PIN)

Configure la broche spécifiée pour qu'elle se comporte soit en entrée, soit en sortie. Les paramètres de cette fonction peuvent être OUTPUT, INPUT, INPUT_PULLUP.

void digitalWrite(Pin, current))

Met un niveau logique HIGH(5V) ou LOW (0V Ground) sur une broche numérique ou analogue. Si le comportement du pin est configuré comme OUTPUT, sa tension sera de 5v⁹ (HIGH) comme sortie ou 0V (LOW).

float analogRead(AnalogPin)

Cette fonction lit la valeur de la tension en entrée présente sur la broche spécifiée, cette valeur peut être entre 0.0049(0V) et 1024¹⁰ (5V).

9. 3.3v sur les cartes qui ont 3.3v comme tension maximale de sortie

10. Les cartes Arduino possède une convertisseur multichannel 10 bit, qui permet de convertir de l'analog vers le digital, donnant ainsi les valeurs mentionnées.

void delay(ms)

Une fonction qui réalise une pause dans l'exécution du programme pour la durée (en millisecondes) indiquée en paramètre. Cette durée doit être de type unsigned long. Les inconvénients de cette fonction sont qu'elle interrompt le programme pendant la durée indiquée en ms, donc toute autre fonction ou opération est interrompue pendant la durée indiquée.

char * sendData(char*, time)

Cette fonction va permettre l'envoi des commandes AT en chaîne de caractères vers le module GSM SIM800L par l'intermédiaire du Moniteur Serial à une vitesse de 9600Bps (Bits par Seconde) et avec un délai en millisecondes comme second paramètre. Ce délai est en fait une attente de réponse du module, si ce dernier ne répond pas dans ce délai alors aucun retour n'est effectué et la chaîne de caractère de réponse sera donc vide.

char* findNumbers(char*)

Cette fonction va permettre de retourner le numéro obtenu lors de l'exécution de la commande sendData(), qui de cette dernière va retourner une réponse de la commande AT qui retourne un numéro avec le nom, le type et le code de région de ce numéro ; Une division de cette chaîne est nécessaire c'est pour cela qu'une recherche doit être effectuée afin de retourner le numéro de la réponse de la commande AT. Il est à préciser que cette fonction aura besoin de la bibliothèque Regexp.h qui est mentionnée antérieurement dans ce mémoire.

2.4 Réalisation du Branchement

2.4.1 Composants nécessaires

Avant de présenter le branchement effectué dans ce travail, nous allons d'abord lister le matériel nécessaire à celui-ci :

- Carte de développement Arduino Uno.
- Prototypage Shield V5.0 ou Breadboard.
- Module MQ-2 ou module MQ-7.
- Le module GSM SIM800L.
- Un buzzer et 2 leds.
- 2 résistances 200 Ohms
- Un relai et un ventilateur.
- Un LCD Crystal I2C (Optionnel).
- Des fils (Jumper).

Remarque 2.4.1 Évitez de toucher les modules dans leur composants (SMD, résistance, etc.), prenez les modules de leurs bords.

2.4.2 Étape 1 : La carte arduino et le prototype shield V5.0

La figure 2.9 montre la carte arduino et le prototype Shield V5.0 avant le branchement, et la figure 2.10 nous montre que le prototype est branché sur l’arduino, les pins indiqués sur le prototype seront donc liés a l’arduino par l’intermédiaire d’un circuit spécifique au prototype shield¹¹.

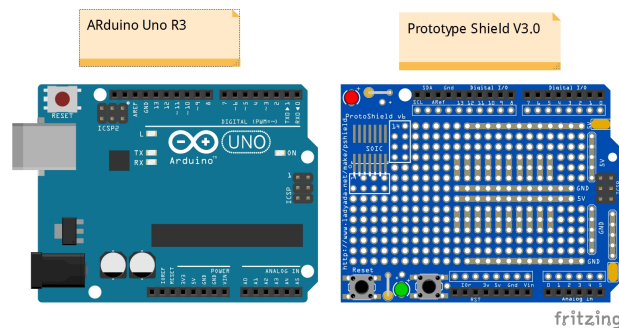


FIGURE 2.9: Arduino and Protoype Shield V5.0 before connection

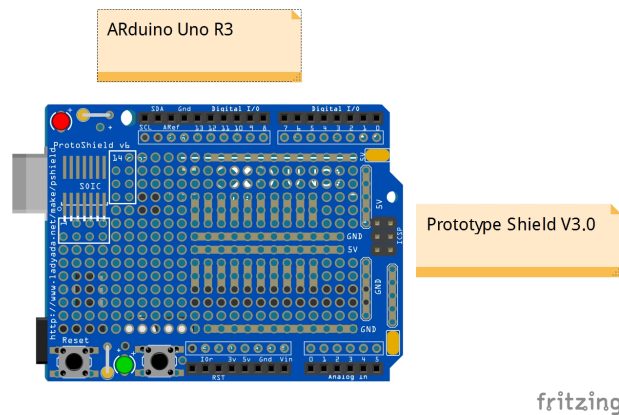


FIGURE 2.10: Arduino and Protoype Shield V5.0 after connection

Remarque 2.4.2 Une alternative au shield serait d'utiliser une breadboard, cependant, sa taille pourrait être encombrante, le système doit être assez compacte.

11. On remarque que l’arduino ne possède pas assez de pins 5V et Gnd, c’est la qu’intervient le prototype shield pour augmenter ce nombre de pins.

2.4.3 Étape 2 : Le Module MQ-2

Son branchement

Après Branchement du Shield, vient le module MQ-2, les pins 5v et Gnd seront branchés sur leurs pins correspondant quant aux pins A0 et D0 du module MQ-2, leur pins seront A0 et D0 respectivement sur le prototype (voir Figure 2.11).

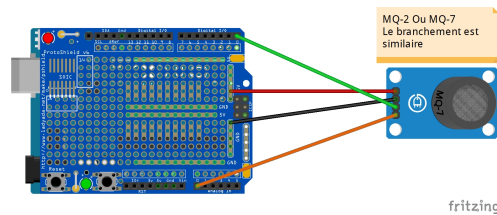


FIGURE 2.11: MQ-2's Connection

Code

```
1 //Initialisation des Module MQ-x
2 pinMode(A0, INPUT);
3 int gassensor;
4 gassensor = analogRead(MQ_PIN); // Return input voltage from 0 to 1023
```

2.4.4 Étape 3 : Les Leds et le Buzzer

Leur branchement

Les Leds et le Buzzer sont les éléments les plus simples dans ce branchement. Le buzzer sera liés à son pin correspondant (pin D4) et le Gnd (Figure 2.12); Quant aux leds, une résistance de 200Ohm¹² sera nécessaire entre leurs pins (pin D5 et D6), pour limiter le courant afin d'éviter d'endommager les leds, et les pins négatifs liés au Gnd (Figure 2.13).

12. La led necessite 30mA(milliAmpere) pour s'allumer correctement, donc $(5V/0.03)=166\text{Ohms}$ ($R=V/I$), une valeur elevé de 20% fera l'affaire.

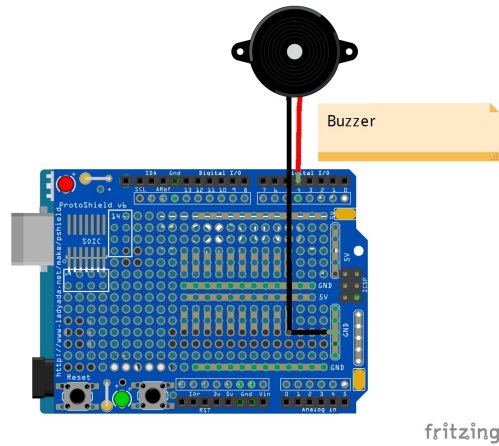


FIGURE 2.12: Buzzer's Connection

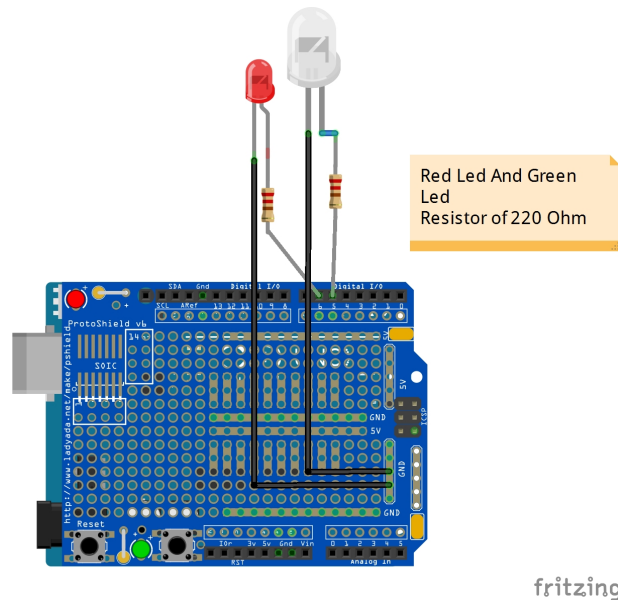


FIGURE 2.13: Led's Connection

Code

```
1 //For Leds
2 pinMode(LED_PIN, OUTPUT);
3 digitalWrite(LED_PIN, HIGH); // turn the LED on (HIGH is the voltage
  level)
4 delay(1000); // wait for a second
5 digitalWrite(LED_PIN, LOW); // turn the LED off by making the voltage
  LOW
6 delay(1000); // wait for a second
7 //For Buzzer
8 pinMode(buzzer_pin, OUTPUT); // Set buzzer - pin 9 as an output
```

```
9  tone(buzzer_pin , 1000); // Send 1KHz sound signal...
10 delay(1000);           // ...for 1 sec
11 noTone(buzzer_pin);    // Stop sound...
12 delay(1000);           // ...for 1sec
```

2.4.5 Étape 4 : Le relai et la ventilation

Leur branchement

Comme expliqué dans le chapitre antérieur, le relai va pouvoir activer des sources d'aération qui utilisent une alimentation plus élevée, son branchement est simple (Figure 2.14) :

- VCC(+)→5V.
- GND(-)→GND.
- Signal(S)→pin D7.

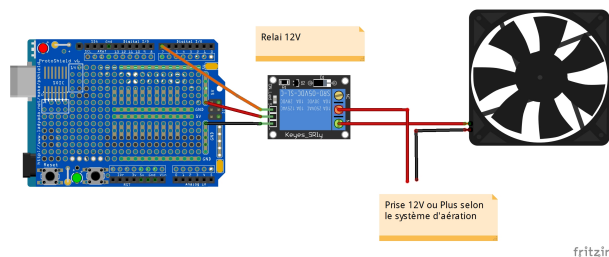


FIGURE 2.14: Relay and Ventilaton's connection

Remarque 2.4.3 *Il Faut faire attention à débrancher le relai de la prise de courant afin d'éviter un choc électrique.*

Code

```
1  //Initialisation du Relai
2  pinMode(RELAY_PIN, OUTPUT);
3  digitalWrite(RELAY_PIN, HIGH); // turn the RELAY on
4  delay(1000);                  // wait for a second
5  digitalWrite(RELAY_PIN, LOW);  // turn the RELAY on
6
```

2.4.6 Étape 5 : Le Module LCD I2C Crystal

Son branchement

Pour le module LCD I2C Crystal, il faut suivre ce branchement :

- VCC→5V.

- GND→GND.
- SDA(Serial Data Line)→pin A4.
- SCL(Serial Clock Line)→pin A5.

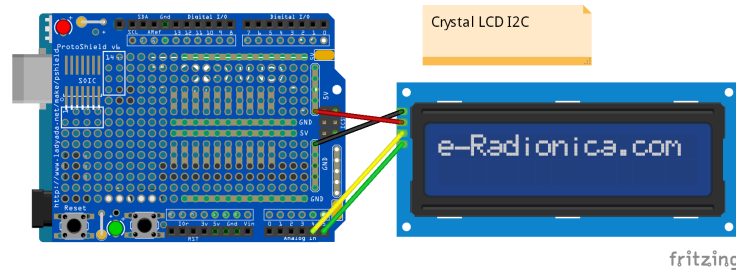


FIGURE 2.15: LCD I2C Crystal's connection

Code

```
1  #include <Wire.h>
2  #include <LiquidCrystal_I2C.h>
3  LiquidCrystal_I2C lcd(0x27,20,4); // set the LCD address to 0x27 for a 16
   chars and 2 line display
4  lcd.init();
5  // Print a message to the LCD.
6  lcd.backlight();
7  lcd.setCursor(0,0);
8  lcd.print("Hello , world!");
```

2.4.7 Étape 6 : Le Module GSM SIM800L

Son branchement

Si ce n'est le dernier et le plus important des modules, le module GSM SIM800L possède un branchement similaire aux autres modules, cependant les pins RX et TX doivent correctement être branchées et doivent, tout comme le module MQ-2, éviter les interférences ou les court-circuits, son branchement est le suivant :

- VCC→5V.
- GND→GND.
- RXD→pin D2;
- TXD→pin D3;

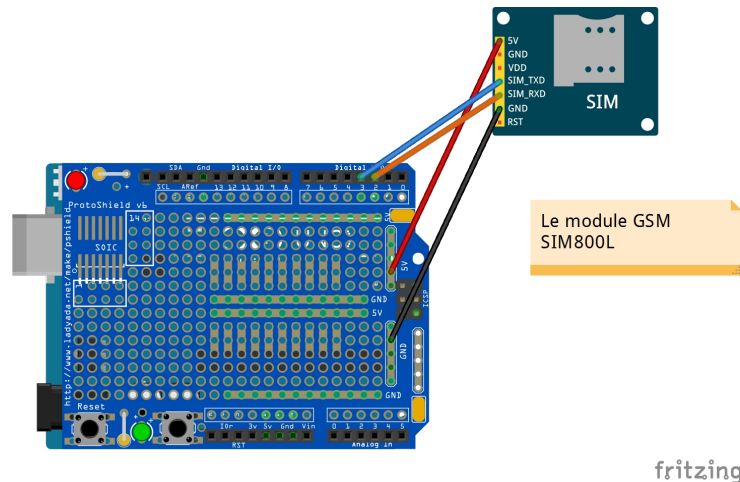


FIGURE 2.16: GSM SIM800L Module's connection

Code

Le code source permettant la communication avec le module GSM SIM800L [14] :

```
1  #include <SoftwareSerial.h>
2
3  //Create software serial object to communicate with SIM800L
4  SoftwareSerial mySerial(3, 2); //SIM800L Tx & Rx is connected to Arduino
   #3 & #2
5
6  void setup()
7  {
8      //Begin serial communication with Arduino and Arduino IDE (Serial
   Monitor)
9      Serial.begin(9600);
10
11     //Begin serial communication with Arduino and SIM800L
12     mySerial.begin(9600);
13
14     Serial.println("Initializing ...");
15     delay(1000);
16
17     mySerial.println("AT"); //Once the handshake test is successful, it will
   back to OK
18     updateSerial();
19     mySerial.println("AT+CSQ"); //Signal quality test, value range is 0-31,
   31 is the best
20     updateSerial();
21     mySerial.println("AT+CCID"); //Read SIM information to confirm whether
   the SIM is plugged
22     updateSerial();
```

```
23     mySerial.println("AT+CREG?"); //Check whether it has registered in the
      network
24     updateSerial();
25 }
26 void loop()
27 {
28     updateSerial();
29 }
30
31 void updateSerial()
32 {
33     delay(500);
34     while (Serial.available())
35     {
36         mySerial.write(Serial.read()); //Forward what Serial received to
      Software Serial Port
37     }
38     while(mySerial.available())
39     {
40         Serial.write(mySerial.read()); //Forward what Software Serial received
      to Serial Port
41     }
42 }
```

2.4.8 Vue Globale des Branchements

La figure 2.17 nous montre le système branché avec le shield, et la figure 2.18 nous montre le branchement sans prototype shield et avec utilisation d'une breadboard.

Remarque 2.4.4 Avec la breadboard, le système est différent et les pins changent.

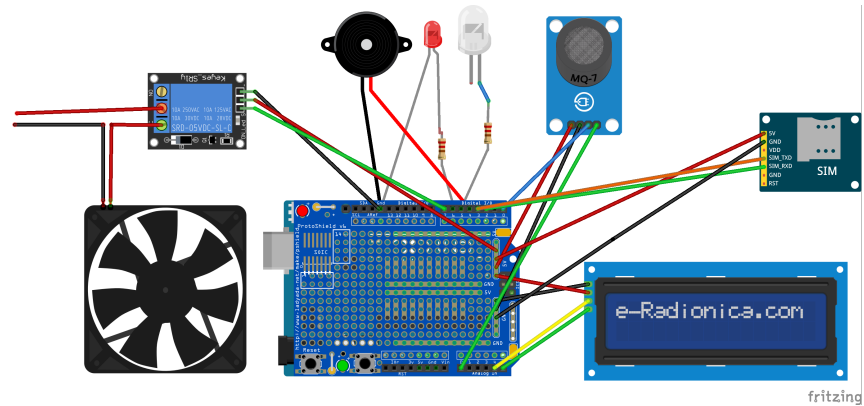


FIGURE 2.17: Full system after connection(without breadboard).

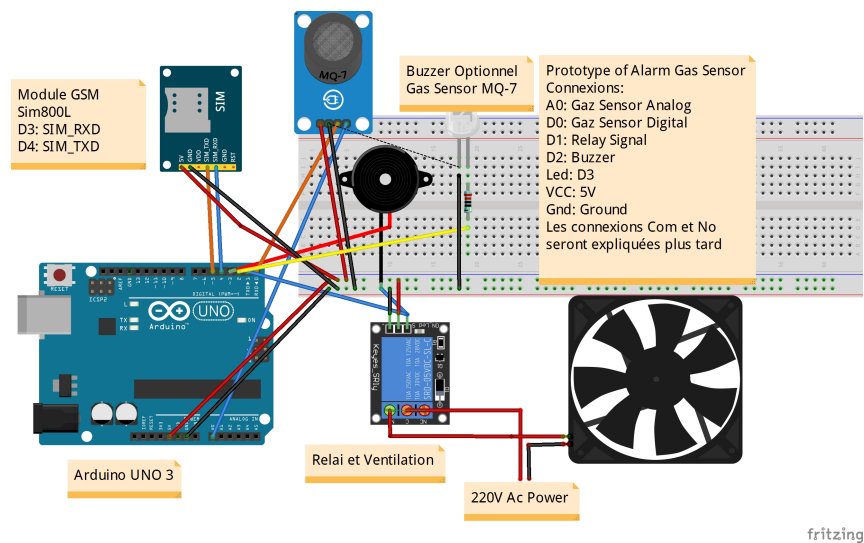


FIGURE 2.18: Full system after connection(with breadboard).

La figure 2.19 nous montre à quoi devrait ressembler le système après branchement (Vue réelle).

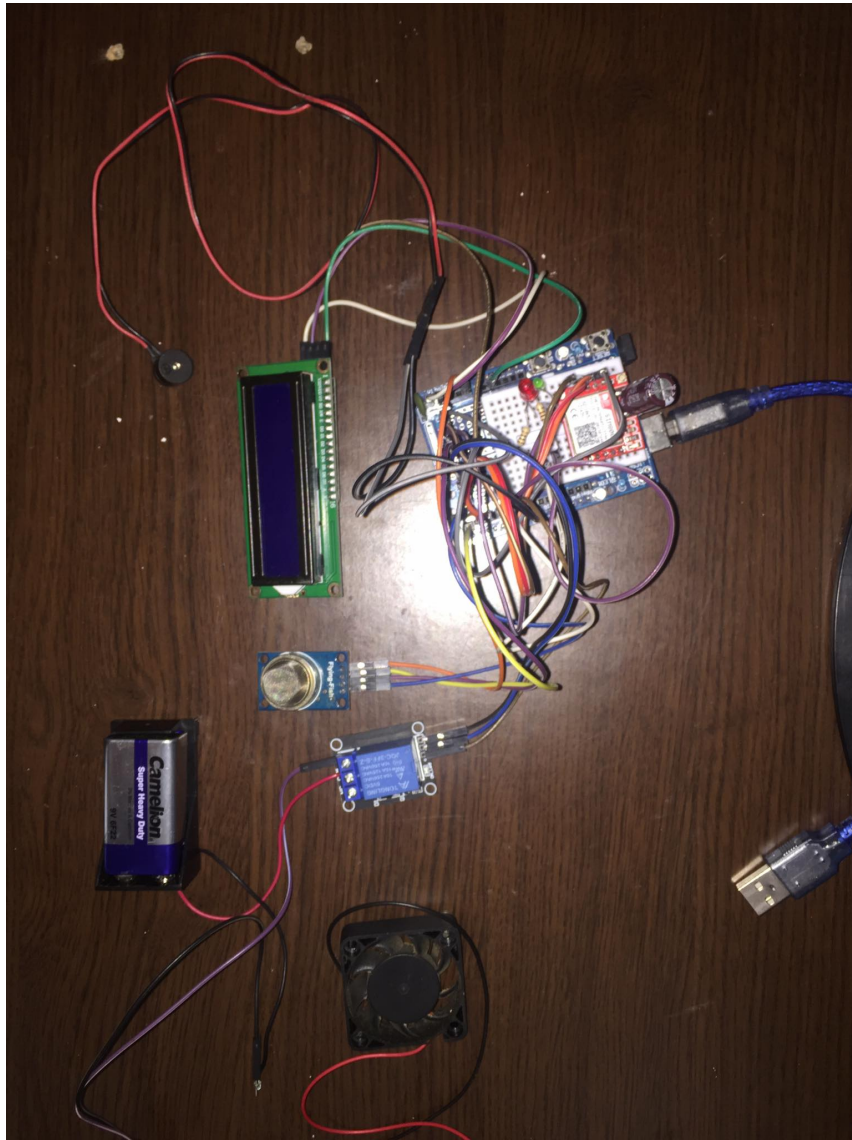


FIGURE 2.19: Le système après branchement (vue réelle)

Après avoir assemblé le système, il faut maintenant télé-verser le code source sur l'arduino, pour cela il faut brancher le câble USB, choisir le type de carte, le port et démarrer le téléversement.

CONCLUSION ET PERSPECTIVES

Conclusion

Durant ces dernières décennies, plusieurs personnes ont trouvées la mort par asphyxie par monoxyde de carbone (CO) en Algérie surtout à l'approche de chaque saison hivernale. Plusieurs raisons sont à l'origine de cette catastrophe, dont on peut citer : la mauvaise aération, la contrefaçon, les mauvaises installations et le non-respect des normes d'entretien. Il est nécessaire que des systèmes de sécurité efficaces soient installés dans des environnements domestiques, industriels et dans des établissements d'éducation.

De nos jours, les systèmes embarqués deviennent de plus en plus présents, et leur utilisation plus fréquentes. Cependant leur utilisation est faite pour des applications natives et simples. C'est pour cela qu'il serait utile d'implémenter une intelligence, même si ce n'est qu'une mineure, et la logique floue est l'une de ces notions qui a permis d'optimiser la façon dont les appareils se comportent.

Ce mémoire a donc été présenté dans le but de créer un système d'alarme afin de notifier de la présence de gas dangereux et à risque. Pour cela, l'utilisation des composants les plus abordables et à faible coût a été abordé, notamment l'arduino, le détecteur de gas MQ-2, le buzzer, les leds, etc.

Perspectives

Nous allons maintenant nous intéresser sur les points qui pourrait être utiles pour des travaux ultérieurs :

1. L'utilisation d'une application de notification par GPRS pourrait être utilisée si les appels échouent.
2. Réduire les dimensions de l'ensemble du système en utilisant une carte PCB et des composants plus performants et moins coûteux.
3. Possibilité d'effectuer une fermeture des vannes de gas afin de prévenir des explosions,(utilisation de servo moteur).
4. Contrôle de la vitesse du système d'aération.

BIBLIOGRAPHIE

- [1] Les systèmes embarqués par Ramzi BOULKROUNE, Mémoire de Master, Memoire-Online, Université de Annaba, 2008, <https://www.memoireonline.com/05/12/5830/Les-systemes-embarques.html>, dernière consultation le 08/07/2021.
- [2] Cédric Locqueneux, Serge Darrieumerlou .« Le guide de la maison et des objets connectés», p31-34.
- [3] Isaed, Walid, Close loop speed control of DC Motor with SCADA system by using Arduino and LabVIEW, Livre Scientifique, 2018.
- [4] UN SYSTEME EMBARQUE POUR LA DETECTION DES GAZ DANGEREUX,Boudjedir Imen, Mémoire de Master en Génie électrique, 2016-2017.
- [5] www.liberte-algerie.com/actualite/alerte-aux-accidents-de-gaz/-85551., dernière consultation le 08/07/2021.
- [6] <https://www.centreantipoisons.be/monoxyde-de-carbone/le-monoxyde-de-carbone-co-en-d-tail/quelles-sont-les-concentrations-toxiques-de>, dernière consultation le 08/07/2021.
- [7] <http://for-ge.blogspot.com/2015/07/microcontroleur.html>, dernière consultation le 8/07/2021.
- [8] <https://www.arduino.cc/en/Guide/Introduction>, dernière consultation le 08/07/2021.
- [9] <https://blog.orsys.fr/les-carnets/index.php/2015/12/18/tour-dhorizon-des-systemes-embarques-linux/> , dernière consultation le 08/07/2021.
- [10] <https://lastminuteengineers.com/mq2-gas-senser-arduino-tutorial/>, dernière consultation le 08/07/2021.

- [11] <https://www.circuito.io/blog/arduino-uno-pinout/>, dernière consultation le 08/07/2021.
- [12] <https://www.microchip.com/wwwproducts/en/ATmega328P>, dernière consultation le 08/07/2021.
- [13] <https://www.indiamart.com/proddetail/jumper-wire-m-m-m-f-f-f-15359553673.html>, dernière consultation le 08/07/2021.
- [14] <https://lastminuteengineers.com/sim8001-gsm-module-arduino-tutorial/>, dernière consultation le 08/07/2021.
- [15] http://www.effled.com/bi-color-led-c-2_42.html, dernière consultation le 08/07/2021.
- [16] <https://visualled.com/fr/glossaire/led-light-emitting-diode/>, dernière consultation le 08/07/2021.
- [17] <https://www.pcboard.ca/minipiezo-buzzer>, dernière consultation le 08/07/2021.
- [18] <https://www.arduino.cc/en/Reference/softwareSerial>, dernière consultation le 08/07/2021.
- [19] <https://osoyoo.com/2018/09/18/micro-bit-lesson-using-the-relay/>, dernière consultation le 09/07/2021.
- [20] <https://www.gotronic.fr/art-capteur-de-gaz-mq5-31498.htm>, dernière consultation le 09/07/2021.
- [21] <https://www.gotronic.fr/art-capteur-de-gaz-mq2-31522.htm>, dernière consultation le 09/07/2021.
- [22] https://fr.hobbytronics.co.uk/mq7-carbon-monoxide-sensor?fbclid=IwAR0ESFsZU4CCbB4W9h6L_ZGSZN8Ys921U5C2m_lOuXR_MuTX-pTgBlHYGo, dernière consultation le 09/07/2021.
- [23] https://www.dzduino.com/index.php?route=product/product&product_id=1943, dernière consultation le 09/07/2021.
- [24] <http://www.elektronique.fr/cours/resistance/resistance.php>, dernière consultation le 09/07/2021.
- [25] <https://www.aspexit.com/la-logique-floue-ou-lextension-de-la-logique-classique/>, dernière consultation le 09/07/2021.
- [26] <https://www.tutorialspoint.com/difference-between-fuzzification-and-defuzzification>, dernière consultation le 09/07/2021.
- [27] <https://www.24pm.com/117-definitions/387-moteur-d-inference>, dernière consultation le 09/07/2021.
- [28] <https://blog.zerokol.com/2012/09/arduino-fuzzy-fuzzy-library-for-arduino.html>, dernière consultation le 09/07/2021.