

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

Université Abou-Bakr Belkaid- TLEMCEM

Faculté de Technologie

Département de Télécommunications

MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : Télécommunications

Spécialité : Systèmes des Télécommunications

Par : Mr. DIDOUH SOUFIANE

Mr. CHOUARI MOHAMMED ELAMINE

Thème

Conception d'un système d'alerte par SMS pour les absences scolaires

Soutenu le : 07- 07 – 2021, devant le jury composé de :

Mr. BENADDA Belkacem	Pr	Université de Tlemcen	Président.
Mr. BAHRI Sidi Mohammed	MCB	Université de Tlemcen	Examineur.
Mr. BORSALI Ahmed Riad	Pr	Université de Tlemcen	Encadreur.

Année universitaire 2020/2021

Remerciements

Nous remercions d'abord le bon Dieu qui nous a donné le courage, la patience, la santé et la volonté continu les études et arriver à la fin de ce travail.

Nos vifs remerciements à notre Encadreur Mr. BORSALI pour son aide, ses conseils, le suivi et l'intérêt qui nous a apporté tout au long de ce travail.

Nous tenons également à remercier chaleureusement les membres du jury Mr. BENADDA. B et Mr. BAHRI S.M qui ont aimablement accepté de juger notre travail et leurs commentaires ont été fructueux.

Merci à la communauté universitaire dans laquelle nous avons vécu pendant cette période et en avons beaucoup appris.

Merci à nos chers amis pour les beaux souvenirs que nous avons eus au cours de ces années.

Dédicace

A mes très chers parents, source de vie, d'amour et d'affection.

A ma petite sœur, source de joie et de bonheur.

A mes grand-mères, source de prière.

A ma petite famille, source d'espoir et de motivation.

A tous mes amis et mes frères, source de souvenirs.

A moi pour la continuité des études et ne jamais abandonner.

A ma première enseignante, elle m'a appris à lire et à écrire.

A vous cher lecteur.

- Didouh Soufiane -

Dédicace

A mes chers parents

Pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études

A mes chères frère et sœurs

Pour leurs encouragements permanents, et leur soutien moral

A toute ma famille

Pour leur soutien tout au long de mon parcours universitaire à tous mes amis Qui m'ont toujours soutenus et encouragés à tous ceux que j'aime et ceux qui m'aime

-Chouari Mohammed Elamine-

Table des matières

Remerciements.....	I
Dédicace.....	II
Table des matières.....	IV
Liste des figures.....	VII
Liste des abréviations.....	IX
Résumé	XI
Introduction générale.....	1

Chapitre I : Etat de l'art de la domotique

I.1 Introduction	3
I.2 Historique de la domotique	3
- Les années 80 : la domotique, signe extérieur de richesse	3
- Les années 1990 : la domotique, en progrès constant	3
- Les années 2000 à ce moment : la domotique est devenue grande	3
I.3 Le principe de fonctionnement	4
I.4 Les fonctions de la domotique	4
A- La fonction de sécurité	5
B- La fonction de surveillance	6
C- La fonction de gestion d'énergie	6
D- La fonction de scénarisation	7
E- La fonction de communication	8
F- La fonction de confort	8
I.5 Les avantages et inconvénients de la domotique ?	9
I.5.1 Les avantages de la domotique	9
I.5.2 Les inconvénients de la domotique	9
I.6 Le prix de l'installation de la domotique doit être en compte	9
I.7 Conclusion	10

Chapitre II : Généralités sur Arduino

II.1 Introduction	11
II.2 L'Arduino	11
II.2.1 Qu'est-ce que le matériel Arduino ?	11
II.2.2 Qu'est-ce que le logiciel Arduino ?	12
II.2.3 quelques types de cartes Arduino :	12
II.2.3.1 La carte Arduino Nano	12
II.2.3.2 La carte Arduino Méga	13
II.2.3.3 La carte Arduino Esplora	13
II.2.3.4 La carte Arduino Micro	14
II.2.3.5 La carte Arduino Lylypad	14
II.2.4 Modules électroniques et Shields	15
II.2.5 Convention de nommage Arduino	16
II.2.6 Que peut-on faire avec un Arduino?	17
II.2.7 Arduino UNO :	18
II.2.7.1 Partie matérielle	18
II.2.7.2 Partie logicielle	23
• Le logiciel Arduino IDE	23
• Le déroulement du programme	24
II.3 La technologie RFID :	25
II.3.1 Les cartes RFID :	25
II.3.1.1 Le Fonctionnement de carte RFID	26
II.3.1.2 Les types des Puces RFID	26
II.3.1.3 A quoi servent les puces RFID ?	27
II.3.2 Le lecteur RFID	27
II.3.2.1 Lecteur RFID : comment ça fonctionne ?	28
II.3.2.2 Choisir un lecteur RFID : fréquence et portée de lecture	28
II.3.2.3 Le module RFID RC522 à 13.56MHz :	28
II.4 le Module GSM/GPRS :	29
II.4.1 SIM900 (GSM/ GPRS) Shield :	29
II.4.2 Vue sur la SIM900	31

II.4.3 Les LED indicatrices	31
II.4.4 AT COMMANDS	32
II.5 Plaque d'essais (Breadboard)	32
II.6 Conclusion	33

Chapitre III : Réalisation d'un prototype de surveillance d'absences.

III.1 Introduction	34
III.2 Le téléchargement et Installation de l'Arduino	34
III.3 Schéma générale de câblage	36
III.3.1 Câblage Arduino + RFID lecteur	36
III.3.2 Câblage Arduino + SIM900	37
III.3.3 Le fonctionnement de circuit au niveau de Arduino IDE	39
III.3.4 Le fonctionnement de circuit au niveau l'interface graphique	40
III.3.4.1 Construire l'interface	40
III.3.4.2 Travaillé avec l'interface	47
III.4 Conclusion	48
Conclusion Générale	49
Annexes	50
Annexe 1 : Programme pour lire le ID de chaque Carte RFID.	50
Annexe 2 : Programme pour Arduino IDE.	52
Annexe 3 : Programme de Visual Basic 6.0.	56
Références Bibliographiques	59

Liste des Figures

Figure I.1: Exemple pour une Maison Intelligente.....	4
Figure I.2: Fonctions de la domotique	5
Figure I.3 : Un exemple pour la fonction de scénarisation	7
Figure II.1 : Quelque type d'Arduino	11
Figure II.2 : Carte Arduino Nano	12
Figure II.3 : Carte Arduino Méga	13
Figure II.4 : Carte Arduino Esplora	13
Figure II.5 : Carte Arduino Micro	14
Figure II.6 : La carte Arduino Lylypad	15
Figure II.7 : Des modules et capteurs peut utiliser avec Arduino	15
Figure II.8 : Différentes Shields peut contrôler avec Arduino	16
Figure II.9 : Les composants de la carte Arduino UNO	18
Figure II.10 : Switch entre alimentation avec USB et Adaptateur DC	19
Figure II.11 : Un microcontrôleur ATmega328P	20
Figure II.12 : Connecteur ICSP	21
Figure II.13 : USB série/alimentation Bridge	22
Figure II.14 : Plateforme de l'IDE Arduino	23
Figure II.15 : Barre de boutons	23
Figure II.16 : Le déroulement du programme	24
Figure II.17 : IDE erreur	24
Figure II.18 : La technologie RFID	25
Figure II.19 : Structure d'un Tag RFID	26
Figure II.20 : Les utilisations de la technologie RFID	27
Figure II.21 : La connexion entre le lecteur et la puce RFID	27
Figure II.22 : Les fréquences et les portées de lecture	28
Figure II.23 : Le module RFID RC522	29
Figure II.24 : SIM900 Shield (GSM/GPRS)	30
Figure II.25 : les composantes de l'extension SIM900	31
Figure II.26 : Schéma de fonctionnement AT COMMANDS	32
Figure II.27 : Les lignes connecter sur la plaque d'essai	33
Figure III.1 : Les systèmes d'exploitation qui prise en charge d'Arduino IDE	34

Figure III.2 : La page de don cliquez « Just Download »	34
Figure III.3 : Les étapes d'installation d'IDE et les portes COM/serial pour Arduino	35
Figure III.4 : L'application IDE été installer (Icon sur le Desktop)	35
Figure III.5 : Schéma général de circuit	36
Figure III.6 : Le brochage de Arduino avec le module RFID RC522	37
Figure III.7 : Switch les cavaliers au (D8, D7)	37
Figure III.8 : Le brochage entre Arduino et SIM900	38
Figure III.9 : Le schéma réel du circuit	38
Figure III.10 : L'élève met sa carte à proximité de lecteur RFID	39
Figure III.11 : Les deux cas de lecteur RFID (connait/ne reconnait pas) la carte RFID	39
Figure III.12 : Fenêtre de nouveau projet	40
Figure III.13 : Fenêtre de l'interface au milieu de fenêtre de projet	40
Figure III.14 : Boite d'outils de Visual basic	41
Figure III.15 : Les labels et ses noms	41
Figure III.16 : Ajouter un grand textBox	42
Figure III.17 : l'emplacement des quatre CheckBox	42
Figure III.18 : La position des trois Timers, un bouton, MsComm1	43
Figure III.19 : Installation de MsComm	43
Figure III.20 : Codage de la Form	44
Figure III.21 : Codage de Timer_1	44
Figure III.22 : Codage de Timer_2	44
Figure III.23 : Codage de Timer_3	45
Figure III.24 : Codage de bouton Command1	45
Figure III.25 : Affichage de sérial moniteur	46
Figure III.26 : Les numéros des parents dans le Croquis	46
Figure III.27 : Une interface simple programmer pour capte les absences	47
Figure III.28 : Fenêtre de rappel à l'enseignant pour envoyer les SMS	47
Figure III.29 : Un exemple d'un SMS recevez par les parents	48

Liste des abréviations

A

AC : Alternative Current.

AREF : Analog Reference.

B

Baud : l'unité de rapidité de modulation en télégraphie.

BF : Basse fréquence.

C

CNC : computer numerical control.

Code ASM : code assembleur.

Croquis : le programme crée.

D

DC : Direct Current.

E

E/S : Entrée / Sortie.

EEPROM : Electrically-Erasable Programmable Read-Only Memory.

G

GPRS : General Packet Radio Service.

GSM : Global System for Mobile Communications.

H

HF : Haute fréquence.

I

ICSP : in-circuit serial programming.

IDE : integrated development environment.

L

LED : light-emitting diode.

M

ME : equipment mobile.

MISO : Master Input, Slave Output.

MOSI : Master Output, Slave Input.

P

PWM : Pulse Width Modulation.

R

RAM : Random Access Memory.

RF : Radio frequency.

RFID : radio frequency identification.

RX : Receiver signals.

S

SCK : Serial Clock, Horloge.

SCL : Signale d'horloge.

SDA : Signal de donnée.

SHF : Super Haute fréquence.

SIM : Subscriber Identity Module.

SMS : Short Message Service.

SPI : Serial Peripheral Interface.

T

TA : Terminal Adaptateur.

TE : Terminal Equipment.

TTL : Transistor-Transistor Logic.

TWI : Two Wire Interface.

TX : Transmitter signals.

U

UART : Universal Asynchronous Receiver Transmitter.

UHF : Ultra Haute fréquence.

USB : Universal Serial Bus.

Résumé :

L'absence scolaire devient un problème de plus en plus important pour les parents, qui doivent être mis au courant en directe d'une possible absence. Ce travail consiste à trouver une solution à ce problème, en munissant d'abord chaque élève d'une carte RFID, détecté à l'entrée de chaque classe. Dans le cas d'une absence, les parents reçoivent une signalisation par SMS, ce qui pourrait les aider à agir directement sans perdre de temps. La réalisation de ce travail est à base d'Arduino, de carte/lecteur RFID et de la puce SIM900.

Mots clés : absence scolaire, RFID, Arduino, SIM900.

Abstract :

Absence from school is becoming an important problem for parents, who must be made aware directly of a possible absence. This work consists of finding a solution to this problem, by first providing each student with an RFID card, detected at the entrance of each classroom.

In case of an absence, the parents will receive an SMS notification, which could help them take direct action without wasting time. The realization of this work is based on Arduino, RFID card / reader and the SIM900 chip.

Keywords : Absence from school, RFID, Arduino, SIM900.

ملخص :

أصبح التغيب عن المدرسة مشكلة ذات أهمية متزايدة بالنسبة للوالدين، حيث يجب أن يكونوا على دراية مباشرة بالغياب المحتمل. يتكون هذا العمل على إيجاد حل لهذه المشكلة، من خلال تزويد كل تلميذ ببطاقة RFID، التي يتم اكتشافها عند مدخل كل قسم. في حالة الغياب، يتلقى الآباء إشعارًا عبر الرسائل القصيرة، والذي يمكن أن يساعدهم في اتخاذ إجراء مباشر دون إضاعة الوقت، يعتمد تحقيق هذا العمل على Arduino و بطاقة / قارئ بطاقة RFID وشريحة SIM900.

الكلمات المفتاحية : التغيب عن المدرسة، RFID، Arduino، SIM900.

Introduction générale

Le système de maison intelligente est un terme utilisé dans la maison, où les appareils connectés au réseau sont utilisés pour fournir une surveillance et une gestion à distance des appareils et des systèmes. La technologie de la maison intelligente, également connue sous le nom de domotique, offre aux propriétaires sécurité, confort et efficacité énergétique en leur permettant de les contrôler via des appareils intelligents (généralement installés sur des smartphones et/ou des tablettes ou d'autres appareils réseau). Les systèmes de maison intelligente tels que les interrupteurs de porte de garage, les systèmes d'alarme et les minuteries de machine à café sont tous des exemples simples de systèmes de maison intelligente.

Cependant, en ce qui concerne la domotique, ces exemples atteindront le plus haut niveau. La maison intelligente, plutôt qu'un dispositif d'exploitation séparé, est contrôlée par l'unité de commande principale de la domotique parmi plusieurs sous-systèmes. Cela signifie que ce module de contrôle automatique principal est similaire à une plaque Arduino, recevant des entrées et des commandes et vérifiant tous les équipements autour de la maison. Ces unités de contrôle exécutent généralement un programme complexe qui leur permet d'effectuer une ou plusieurs actions en fonction de différents événements. Ces événements peuvent prendre plusieurs formes, mais seules deux catégories peuvent être fondamentalement séparées : le temp et l'activation de l'événement.

L'un des problèmes actuels est que les enfants sont absents de l'école à l'insu de leurs parents. Cela a causé l'échec scolaire, puis a été expulsé. Cela effacera sa moralité et tout ce qu'il a enseigné à l'école, et créera une catégorie sociale caractérisée par l'ignorance, qui est considérée comme l'ennemi de la science et du progrès. Il existe aussi la possibilité que l'enfant n'atteigne pas l'école et qu'il soit enlevé, dans ce cas les parents sont directement alertés par SMS.

Dans le cadre de nos recherches, nous avons mené des recherches pour utiliser la technologie pour trouver une solution à ce problème. La solution est de créer un système automatique pour l'alerte au niveau de l'école.

Au cours du premier chapitre, nous avons défini le rôle de la domotique et ses fonctionnalités, en la rendant plus conviviale car elle se connecte à tous les appareils qu'elle contient.

Dans le deuxième chapitre, nous avons expliqué les composantes d'un système Arduino avec certaines de leurs caractéristiques, et mentionné quelques versions d'Arduino.

Enfin le troisième chapitre, est consacré à la réalisation pratique du système avec un système de marquage des absences scolaire.

Nous considérons l'automatisation comme une solution utile et claire dans tous les domaines. Le succès de nos recherches nous a amenés à accorder plus d'attention à l'automatisation et aux méthodes que nous pouvons l'utiliser pour éliminer tous les problèmes surtout les fléaux sociaux.

Chapitre I :
Etat de l'art de la domotique

I.1 Introduction :

Du latin « domus » signifie que la domotique de la maison est électronique, informatique et télécommunications pour automatiser les bâtiments individuel ou collectif.

La domotique est un terme général qui regroupe toutes les technologies utilisées pour automatiser la gestion d'une maison. Ces technologies font un usage intensif des ordinateurs et des objets de La connexion facilite la vie des utilisateurs.

Cet ensemble de techniques peut servir à plusieurs choses : automatiser certaines tâches pour faciliter la vie de l'utilisateur, augmenter la sécurité de l'habitat, ou encore réduire la consommation énergétique du logement.

I.2 Historique de la domotique :**- Les années 80 : la domotique, signe extérieur de richesse.**

Soutenue par une exigence accrue des ménages en termes de confort, la domotique fait véritablement son apparition dans les années 80. Les avancées dans les domaines de l'électronique (miniaturisation des composants), de l'informatique (arrivée des premiers ordinateurs dans les foyers) et de la communication (numérisation des réseaux, minitel) sont à l'origine de cette nouvelle technologie.

- Les années 1990 : la domotique, en progrès constant

La maison en 1990 est automatisée : la télécommunication commence à se développer. Les réseaux informatiques deviennent sans fil grâce au wifi. Certaines actions sont planifiées (ouverture des volets à une heure donnée, allumage des lumières quand on passe à proximité ou quand le soleil se couche...), les télécommandes remplacent les interrupteurs, et les alarmes s'associent à la télésurveillance.

- Les années 2000 à ce moment : la domotique est devenue grande !

Le secteur de la domotique n'a cessé de croître depuis 2000. La maison peut désormais être contrôlée à distance ! Avec le développement d'Internet, l'automatisation est programmée et contrôlée via un appareil mobile (Smartphone, tablette, ordinateur). La maison est soumise et les solutions « sur mesure » permettent de proposer un écosystème de produits parfaitement adapté aux besoins du client. Autrement dit, la domotique se démocratise, avec l'aide de professionnels de la construction (architectes, artisans, ingénieurs thermiques, etc.) qui sont plus susceptibles d'être formés à fournir des conseils et des compétences à leurs clients.[1]

I.3 Le principe de fonctionnement :

Le principe de la domotique est de faire communiquer les appareils entre eux. Equipement électronique du bâtiment. Par conséquent, nous parlons de bâtiments intelligents ou de bâtiments de communication.

La domotique peut superviser et programmer les fonctions du bâtiment afin de répondre. Répondez à vos attentes en matière de sécurité, de confort, de gestion de l'énergie et de communication. Elle a participé les personnes âgées ou handicapées en favorisant le maintien à domicile du quotidien.

Cette communication est possible grâce aux différents outils de pilotage dont disposent les utilisateurs. On dénombre :

- Les ordinateurs centraux ou de poches,
- Les téléphones portables ou Smartphones (commande à distance ou sur place),
- Les télécommandes (universelles ou non),
- Les écrans tactiles...

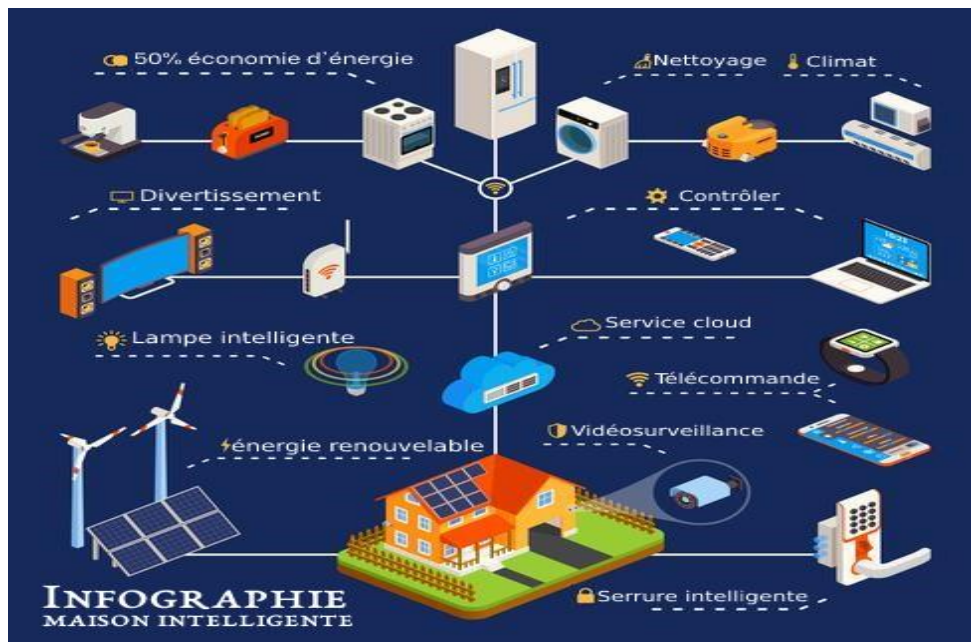


Figure I.1 : Exemple pour une Maison Intelligente

I.4 Les fonctions de la domotique :

Les fonctions suivantes peuvent être réalisées grâce aux technologies intégrées dans la domotique (voir figure I.2) :

- Sécurité.
- Surveillance.
- Gestion de l'énergie.
- Scénarisation des actions.
- Communication.
- Confort.

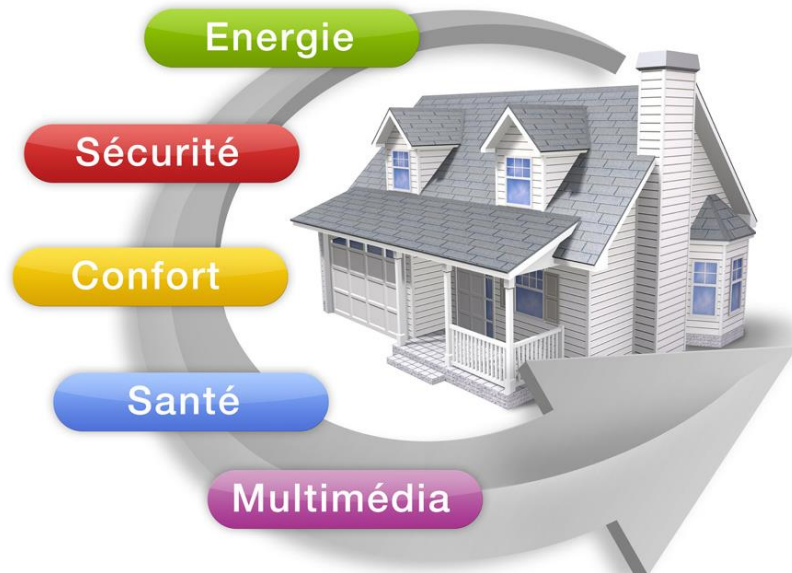


Figure I.2 : Fonctions de la domotique

A. La fonction de sécurité :

La sécurité est aussi avant tout un point recherché. Quand on quitte son domicile on souhaite qu'elle reste surveillée et grâce aux technologies qui sont devenues grand public c'est maintenant à notre disposition. On cherche à éviter les intrusions, les tentatives de violations, intempéries (même légères).

Contrôler et enregistrer l'accès à la maison ou à l'entreprise, en cas d'intrusion, après la détection, la centrale prend les mesures nécessaires :

- Simulez l'intrus en allumant les lumières au hasard pour empêcher l'intrus, Radio ou ouvrir les stores pendant la journée, ...
- Dissuasion (sirène)
- Surveillance de l'accès par des pièces du logement (caméras externes et internes)
- Transmission et visualisation des alarmes sur incident (envoi de mail) [2]

A.1 Les alarmes techniques :

Le système domotique peut vous alerter en local ou à distance en cas d'anomalies des équipements suivants :

- Congélateur à l'arrêt ;
- Détection fuite de gaz ;
- Fuite d'eau ;
- Incendie ;
- Fumée.

A.2 Simulation de présence en cas d'absence :

En programmant le système de sorte que les volets se baissent et se relèvent à des heures déterminées, ou pour que les lumières s'allument et s'éteignent, afin de donner l'impression que votre maison est occupée même si vous êtes partis en vacances.

A.3 Les vidéosurveillances :

Les caméras installées à l'extérieur et à l'intérieur de l'habitation permettent d'observer tout ce qui se passe dans et autour de la maison et suivre les anomalies en direct. Sur détection de présence, la vidéo est envoyée par mail au correspondant.

B. La fonction de surveillance :

La domotique peut proposer un panel très large de détecteurs et de capteurs : des mouvements, des inondations, de bris de verre, des vibrations, d'ouverture de porte et fenêtre, de fumée / feu, d'humidité / inondation, de pluie, le vent, la tempête...

Si quelqu'un est présent, la centrale intervient immédiatement pour couper le courant, ouvrir les stores, couvrir la piscine, faire un appel d'urgence ou déclencher l'alarme. [3]

C. La fonction de gestion d'énergie :

Gérer l'énergie domiciliaire est une des préoccupations de notre société actuelle, vue l'augmentation des coûts de l'énergie. La domotique est une technologie qui apporte à la gestion de l'énergie les outils qui lui manquaient.

En effet, La domotique peut offrir la réduction des consommations énergétiques des bâtiments en adaptant ces consommations aux modes de vie des occupants et à l'environnement extérieur. Cela

comprend la régulation de l'éclairage et du chauffage, le 17 traitement de l'air, l'optimisation des ouvrants, la programmation horaire, les commandes à distance, les interrupteurs automatiques pour l'éclairage d'un escalier ou d'un couloir, l'ouverture ou la fermeture d'un volet selon l'ensoleillement etc...

Il prend en compte la présence des occupants et s'adapte automatiquement à leur horaire :

- Éteignez les lumières inutiles
- Ajuster l'intensité lumineuse en fonction de l'activité

Un simple coup de fil suffit pour augmenter la température du chauffage en cas d'arrivée des occupants.[2], [3]

D. La fonction de scénarisation :

Une seule commande permet de valider plusieurs fonctions entièrement personnalisées ou à personnaliser, chaque fois que cela s'avère nécessaire. Par exemple, le scénario de vie SORTIR peut prévoir l'extinction de l'éclairage, la fermeture des volets roulants, le réglage du chauffage et l'activation de l'alarme. L'option SIMULER PRÉSENCE peut prévoir l'ouverture des volets et l'allumage des lumières exactement comme si vous vous trouviez chez vous. Avec la domotique, les scénarios peuvent également être commandé sa distance (fonction de contrôle à distance sur PC, tablette ou smartphone).[4]

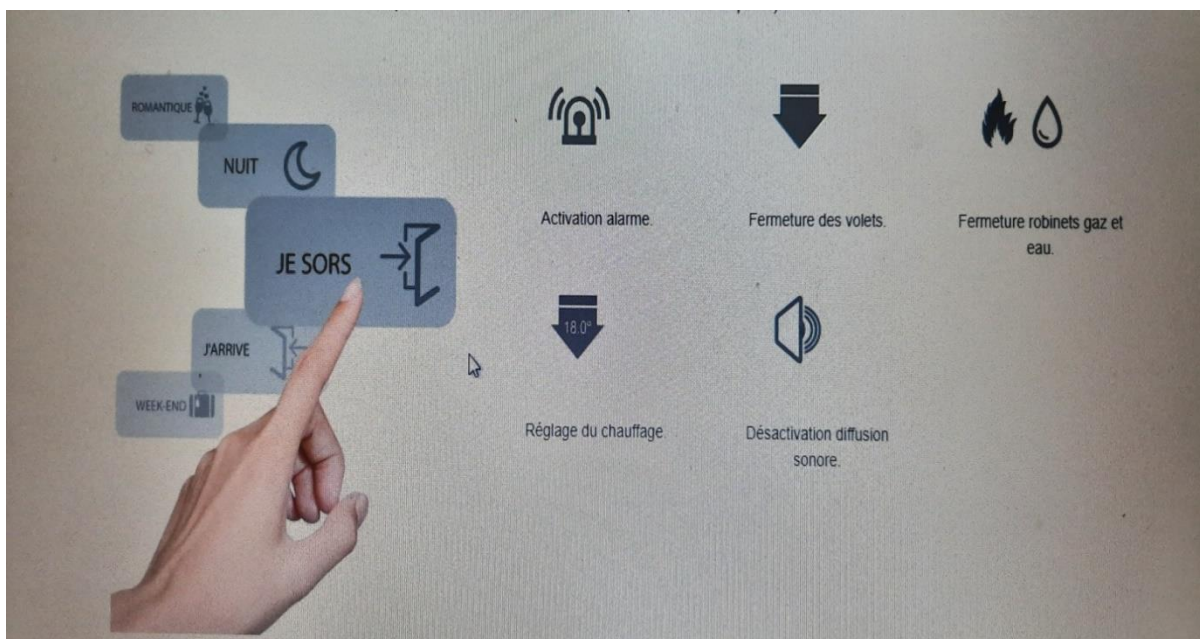


Figure I.3 : Un exemple pour la fonction de scénarisation

E. La fonction de communication :

Un système domotique permet la communication non seulement à l'intérieur de la maison, mais aussi à l'extérieur. Il n'est pas obligatoire d'être à la maison pour commander les appareils domotique car il suffit d'exploiter les réseaux comme internet par exemple. Grâce à la centrale domotique connectée à internet on peut recevoir l'état de son installation et d'émettre des alertes et piloter sa maison de n'importe quel endroit du monde, de son bureau ou de sa voiture.

La domotique permet à l'utilisateur de communiquer avec son domicile via :

- Par téléphone (GSM)
- Par ordinateur (Internet) [2], [3]

F. La fonction de confort :

L'homme veut toujours la facilité. De ce fait, certaines fonctions dans notre habitation peuvent être amélioré ou automatisé pour avoir un confort.

De l'émetteur radio par un télécommande infrarouge, un téléphone, une horloge ou même un détecteur sont les différents dispositifs de commande pouvant agir sur les appareils domotique et un même détecteur peut engendrer des actions différentes, par exemple :

- Réglage de l'intensité lumineuse suivant l'activité. Lorsque vous rentrez le soir, d'un simple geste le portail s'ouvre et l'éclairage extérieur s'allume
- Enclenchement automatique de certains luminaires en fonction des détecteurs de présence.
- Scénarios d'ambiance modifiables : lumières, chauffages, ouvrants, etc.
- Diffusion multimédia (TV, Téléphone, Informatique) via prises de communication universelles RJ45.
- Diffusion sonore Multi room.
- Commande des stores et persiennes depuis le panneau de commande ou en fonction de la lumière extérieure.
- Arrosage automatique de la pelouse suivant le détecteur d'humidité (pas d'arrosage les jours de pluie)
- Gestion du système d'antigivrage
- Gestion du jardin d'hiver (maintien de la température, de l'humidité, de la densité d'éclairage)

- Gestion de la piscine (maintien de la piscine à température constante et suivi de la qualité d'eau) [3],[5]

I.5 les avantages et inconvénients de la domotique ?

La domotique est une solution très populaire pour automatiser nos maisons. De plus, qu'il s'agisse de sécurité ou de travaux ménagers, cela nous facilite la vie au quotidien.

I.5.1 Les avantages de la domotique :

- Plus de confort grâce à la domotique. La domotique augmente considérablement votre confort de vie.
- Davantage de contrôle sur votre habitation grâce à la domotique. ...
- Économiser de l'énergie et de l'argent grâce à la domotique. ...
- Décentralisation.
- Chronophage

I.5.2 Les inconvénients de la domotique :

- Piratage : hackage facile du Smartphone et accès facile aux informations personnelles.
- Vol/perde/dépendance : en cas de perte du Smartphone, incapacité d'utilisation du système.
- Déchets : beaucoup plus de déchets électroniques à traiter.
- Coût : installation onéreuse.
- Consommation d'électricité : consommation élevée de certains appareils et néfaste pour l'environnement. [6]

I.6 Le prix de l'installation de la domotique doit être en compte :

- L'achat et l'installation de la centrale domotique.
- L'achat et l'installation des éléments pilotés par le système, ceux-ci devant être compatibles entre eux.
- L'installation du système qui relie les différents éléments : Bus de commande, sur courant porteur ou sans fil.
- La configuration du service qui peut être faite par l'utilisateur ou par un domoticien, en fonction de la complexité du système.

I.7 Conclusion :

Ce chapitre est consacré à un état de l'art de la domotique avec tous ces principes et caractéristiques, allant de la sécurité, confort, alarme... Nous avons constaté qu'elle devient de plus en plus importante dans la vie de tous les jours.

Le chapitre suivant est consacré aux composant utiles dans la domotique et notamment la carte Arduino.

Chapitre II :
Généralités sur Arduino

II.1 Introduction :

Avant de nous entraîner pour réaliser notre premier projet, nous devons d'abord parler des composantes de ce projet qui contient la Sim900 et RFID (lecteur et puce), qui sont contrôlés par la carte Arduino, et aussi nous apprendrons à connaître la famille Arduino et certaines plaques les plus célèbres, l'environnement de développement de la plaque 'Arduino IDE' et le déroulement de programme.

II.2 L'Arduino :

Arduino est un circuit imprimé, un ensemble de matériel et de logiciels sur lequel se trouve un microcontrôleur qui vous permet de vous familiariser avec la programmation informatique et d'apprendre avec les appareils électroniques.

Arduino est open source. Ainsi, vous pouvez télécharger le design original et préparer et vendre votre carte sans payer de droits d'auteur. La figure II.1 présente quelques prototypes d'Arduino.

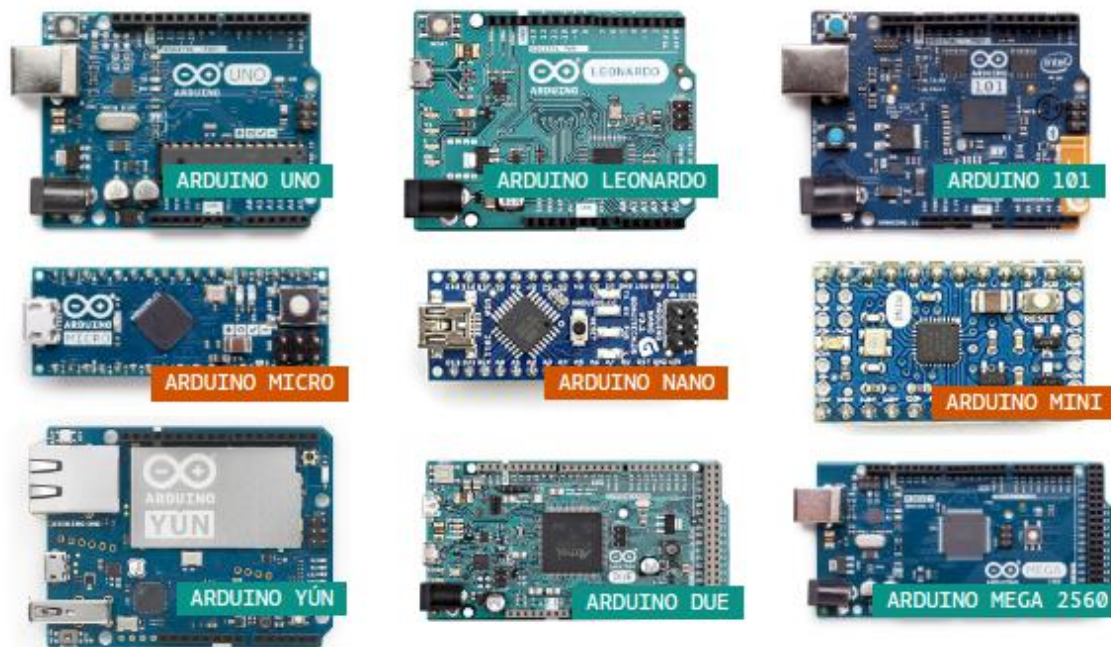


Figure II.1 : Quelques type d'Arduino.

II.2.1 Qu'est-ce que le matériel Arduino :

Ce sont des cartes électroniques programmables (y a un processeur et une mémoire), nous pouvons brancher : des LED, des capteurs de température, des caméras, de vibrations ou de lumière,

d'humidité, des boutons, des potentiomètres, des contacts électriques, des moteurs, des relais, des moniteurs.

La carte Arduino est un cerveau qui active les systèmes électroniques intelligents et les dispositifs mécaniques.

II.2.2 Qu'est-ce que le logiciel Arduino :

Les créateurs de Arduino ont développé un software qui permet de programmer les cartes C'est ce que l'on nommé L'IDE Arduino, un Environnement de Développement « Intégré » qui permet de crée et configurer votre programme, l'enregistrer, le compiler, le vérifier, le transvaser sur une plaque Arduino ... [7]

II.2.3 Quelque type de cartes Arduino :

II.2.3.1 La carte Arduino Nano :

Le modèle Arduino Nano est une carte électronique dont le cœur est un microcontrôleur ATMEL, référez-vous à ATmega328. Microcontrôleurs de la série AVR pouvant être programmés en langage Arduino C. La carte dispose de 14 entrées/sorties numériques (dont 6 utilisables en PWM (Pulse Width Modulation), 6 entrées analogiques avec convertisseur analogique/numérique résolution 10 bits, 1 résonateur céramique (quartz), 16 Mhz. 1 connecteur ICSP (programmation série en circuit) permet de programmer le microcontrôleur sur le circuit sans le retirer, 1 connecteur jack pour alimentation externe et un bouton reset pour le processus de reset, L'avantage de cette carte est qu'elle n'a pas besoin de driver pour faire de l'USB /conversion série.

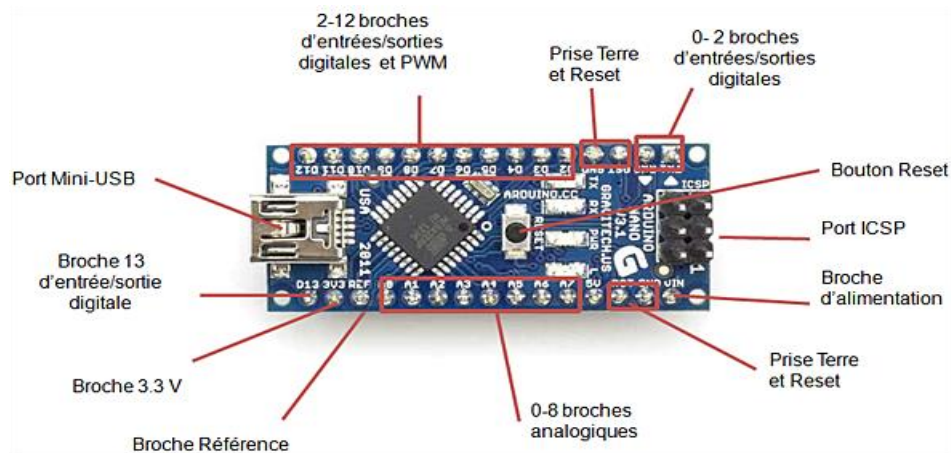


Figure II.2 : Carte Arduino Nano.

II.2.3.2 La carte Arduino Méga :

Arduino Mega 2560 (Figure II.3) est une carte microcontrôleur basée sur ATmega2560. Il dispose de 54 broches d'entrée/sortie numériques (dont 15 peuvent être utilisées comme sorties PWM), 16 entrées analogiques, 4 UART (ports série matériels), un oscillateur à cristal de 16 MHz, une connexion USB, une prise d'alimentation et une connexion ICSP. Appareil et un bouton de réinitialisation. Il contient tout le nécessaire pour prendre en charge le microcontrôleur ; il suffit de le connecter à l'ordinateur avec un câble USB ou d'utiliser un adaptateur AC-DC ou une batterie pour l'alimenter et commencer à l'utiliser.

La carte Mega 2560 est compatible avec la plupart des shields conçus pour Uno et les anciennes cartes Duemilanove ou Diecimila. [8]



Figure II.3 : Carte Arduino Méga.

II.2.3.3 La carte Arduino Esplora :

La carte Esplora (Figure II.4) est dérivée de la carte Arduino Leonardo et est basée sur ATmega 32U4. La disposition de la carte est similaire à celle d'une manette de jeu et est équipée de capteurs et d'actionneurs, qui peuvent être utilisés directement pour les tests.

Esplora est conçu pour les personnes qui souhaitent utiliser des modules Arduino mais ne comprennent pas l'électronique. Il peut être programmé avec le logiciel Arduino. Le contrôleur ATmega32u4 permet la gestion des ports, ce qui augmente la flexibilité de la communication avec l'ordinateur. [9]

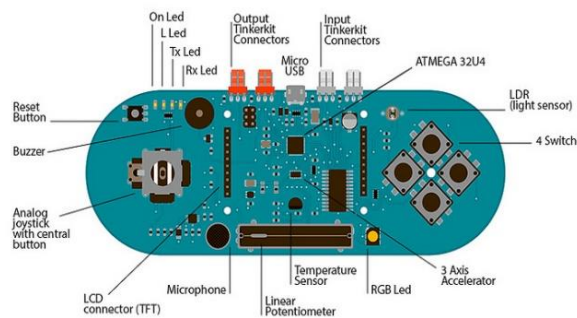


Figure II.4 : Carte Arduino Esplora.

II.2.3.4 La carte Arduino Micro :

La carte Arduino Micro (figure II.5) est basée sur ATmega32u4 avec une fréquence d'horloge de 16 MHz, permettant à un seul processeur de gérer le port USB. Il dispose de 20 broches d'E/S et d'un connecteur micro USB, qui peut être branché sur une carte de test sans soudure. La carte se compose de 2 x 17 contacts au pas de 2,54 mm, permettant une intégration facile de la carte dans votre projet Arduino.

Le contrôleur ATmega32u4 permet la gestion du port USB, augmentant ainsi la flexibilité de communication avec l'ordinateur. [10]

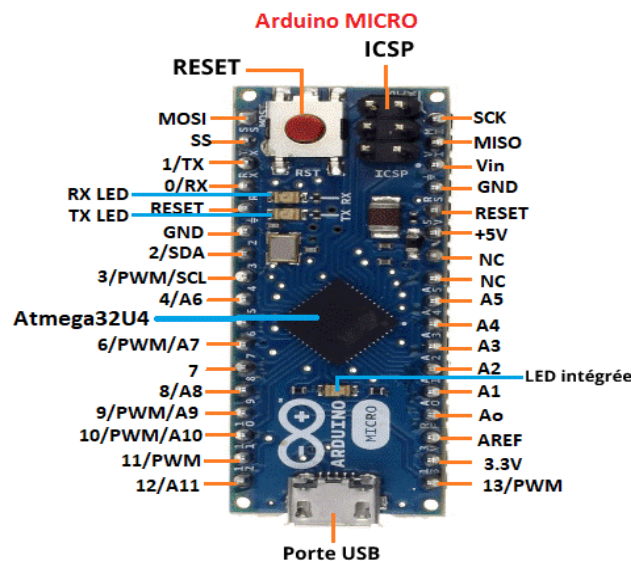


Figure II.5 : Carte Arduino Micro.

II.2.3.5 La carte Arduino Lylypad :

Le système LilyPad (figure II.6) est un écosystème complet, comprenant un ensemble de composants de couture électroniques conçus pour développer des vêtements interconnectés, dynamiques et interactifs.

Les modules tels que les LED, les buzzers, les vibrateurs, les capteurs, les moteurs et les modules Xbee sont connectés au microcontrôleur LilyPad à l'aide de fils à coudre conducteurs.

Le module LilyPad peut obtenir des informations de l'environnement extérieur et activer des LED, buzzers, haut-parleurs, moteurs, etc. La LilyPad a été conçue par Leah Buechley en collaboration avec SparkFun [11]

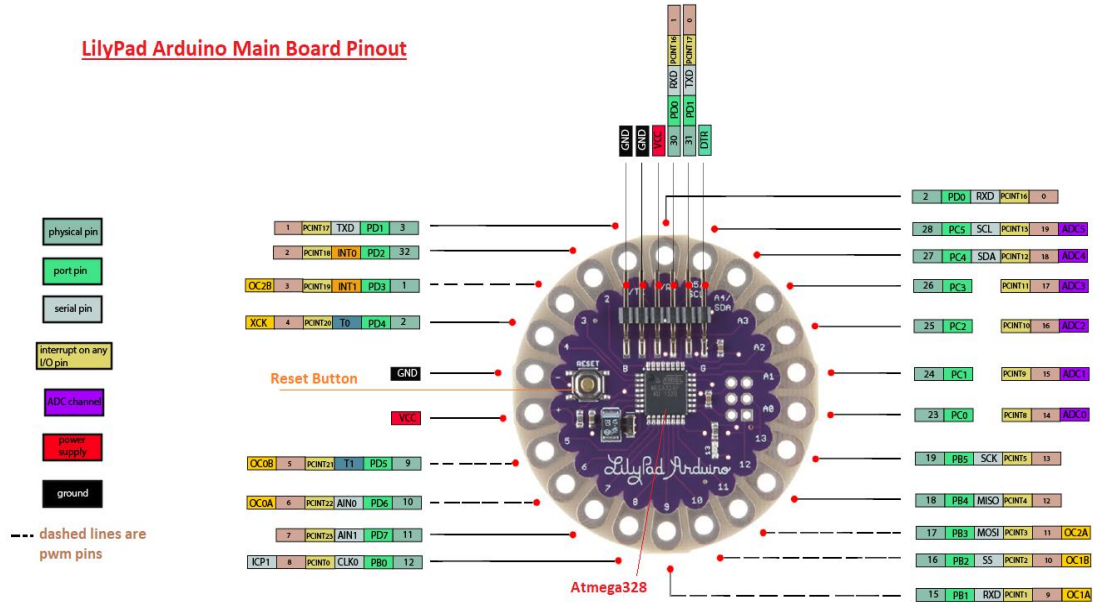


Figure II.6 : La carte Arduino Lylypad.

II.2.4 Modules électroniques et Shields :

1-Modules :

Arduino peut utiliser diverses mini-cartes. Cela facilite le test de composants spécifiques avec des cartes précâblées, tels que des capteurs de température, des modules de détection par ultra-sons, des écrans LED, etc.

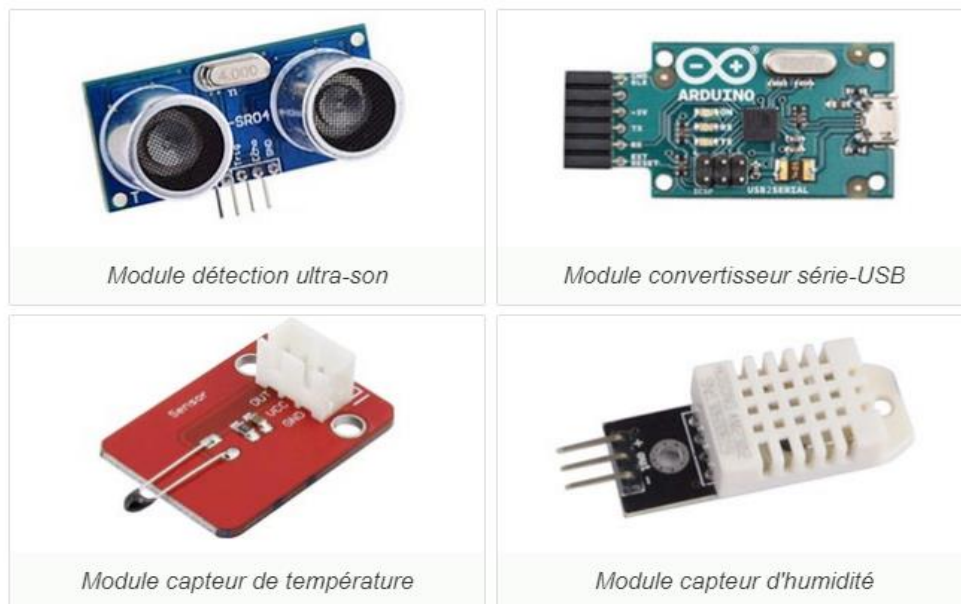


Figure II.7 : Des modules et capteurs peut utiliser avec Arduino.

2-Shields :

La couche de blindage est une carte électronique compatible broche à broche avec un connecteur de carte Arduino. Ce sont de véritables extensions qui améliorent les fonctionnalités de l'Arduino de base (figure II.8).

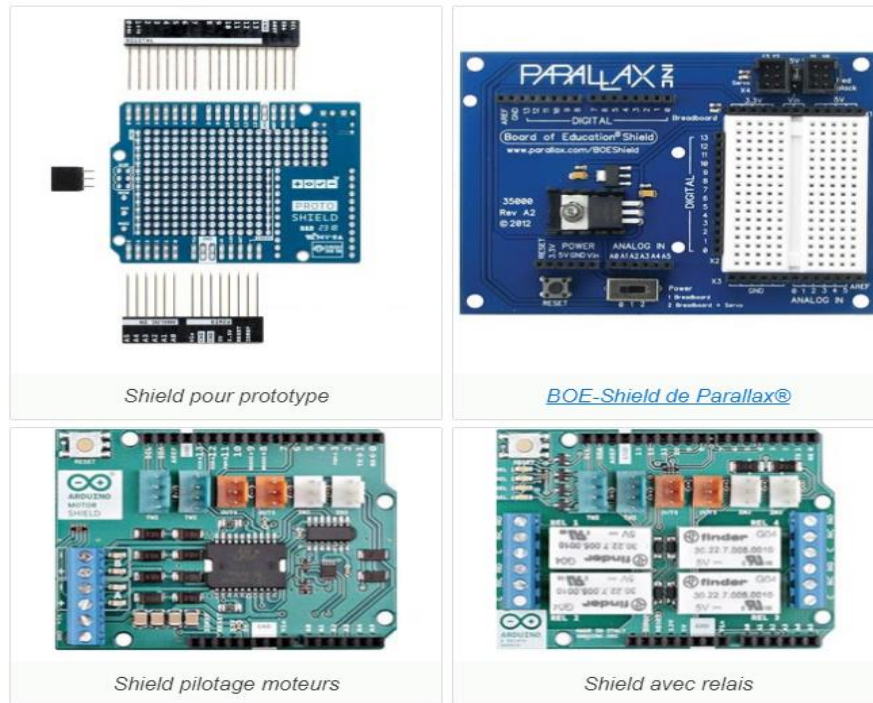


Figure II.8 : Différentes Shields peut contrôler avec Arduino.

II.2.5 Convention de nommage Arduino

Bien que la conception du circuit Arduino et de son logiciel soit open source, l'équipe Arduino a réservé le terme « Arduino » pour sa propre conception. Le logo Arduino est une marque déposée.

Parfois, vous constaterez que cela ressemble à une carte Arduino officielle mais qu'il ne s'agit pas d'un produit fabriqué par l'équipe Arduino. Certains fabricants utilisent "-duino" ou "-ino" dans les noms de produits, tels que Freeduino, Funduino, Diavolino, Youduino, etc. Certains produits (tels que les cartes fabriquées par SainSmart) n'utilisent que des noms de modèles (tels que Uno et Mega2560).

En raison du différend entre la société créée par le fondateur d'origine (Arduino LLC) et l'un des fondateurs d'origine (Arduino SRL), Arduino LLC utilise la marque Arduino aux États-Unis et Genuino ailleurs.

Certains "fabricants" prétendent vendre des cartes Arduino, mais ils utilisent en fait des copies de la marque sans autorisation. Massimo Banzi a spécifiquement introduit ces copies non autorisées dans son blog.

Le point de base à retenir ici est que vous pouvez copier le schéma, le code du chargeur de démarrage, le code IDE Arduino et les utiliser pour créer votre propre version (principe open source, veuillez ne pas l'appeler "Arduino").

II.2.6 Que peut-on faire avec un Arduino ?

En plus de l'IDE Arduino facilitant la programmation, une autre fonction importante d'Arduino est la capacité du microcontrôleur sur lequel il est basé.

Avec des blindages plus faciles à utiliser, une variété de modules de capteurs et d'actionneurs économiques, vous pouvez en effet faire beaucoup de choses avec Arduino.

La condition est de garder à l'esprit certaines restrictions de base : mémoire, fréquence d'horloge, courant de sortie périphérique et niveau de tension.

-Mesure et détection :

- Station météorologique automatisée,
- Détecteur de foudre,
- Suivi du soleil pour orientation des panneaux solaires,
- Moniteur de radiation,
- Détecteur automatique de la faune,
- Système de sécurité domestique ou professionnel.

-Contrôle :

- Petits robots,
- Maquette de fusée ou d'avion,
- Drones multi-rotor,
- CNC simple pour petites machines-outils.

-Automatisation :

- Serre automatisée,
- Maison automatisée,
- Aquarium automatisé,
- Robot navette d'échantillon de laboratoire,
- Enceinte thermique de précision (couveuse, yaourtière, étuve, séchoir...),
- Système de test électronique automatisé.

-Art :

- Contrôle d'éclairage et sonore dynamique,
- Structures cinématiques,
- Œuvre d'art,
- Cinéma 3D avec les Drones. [7]

II.2.7 Arduino UNO :

II.2.7.1 Partie matérielle

Il n'y a pas de modèle unique de carte Arduino, il existe de nombreuses variantes d'Arduino, chacune étant conçue pour répondre à des objectifs différents. Choisir la bonne carte n'est pas toujours facile, car le nombre de cartes ne cesse d'augmenter. Cependant, la carte peut être considérée comme la pierre angulaire de toute aventure dans le monde Arduino ; c'est celle que la plupart des gens commencent à utiliser et qui convient à la plupart des usages. C'est l'Arduino (ou Genuino) Uno. La dernière version est toujours Arduino Uno R3 (sortie en 2011). Il s'agit d'un matériau de haute qualité et fiable adapté à de nombreux projets. Si vous débutez, cette carte est faite pour vous.

Uno est le mot italien pour le chiffre un, et correspond ici à la version 1.0 du logiciel Arduino. Les prédécesseurs ont pris divers noms, tels que Serial, NG. Uno est intervenu et a ordonné le nom des cartes. R3 fait référence à l'examen des fonctions de la carte, qui est un ensemble de mises à jour, d'améliorations et de réparations. Par conséquent, dans ce cas, il s'agit de la troisième révision. En plus du microcontrôleur, la carte comprend également de nombreux petits composants, qui seront décrits dans ce chapitre. La figure II.9 montre les composants de la carte Arduino UNO. [12]

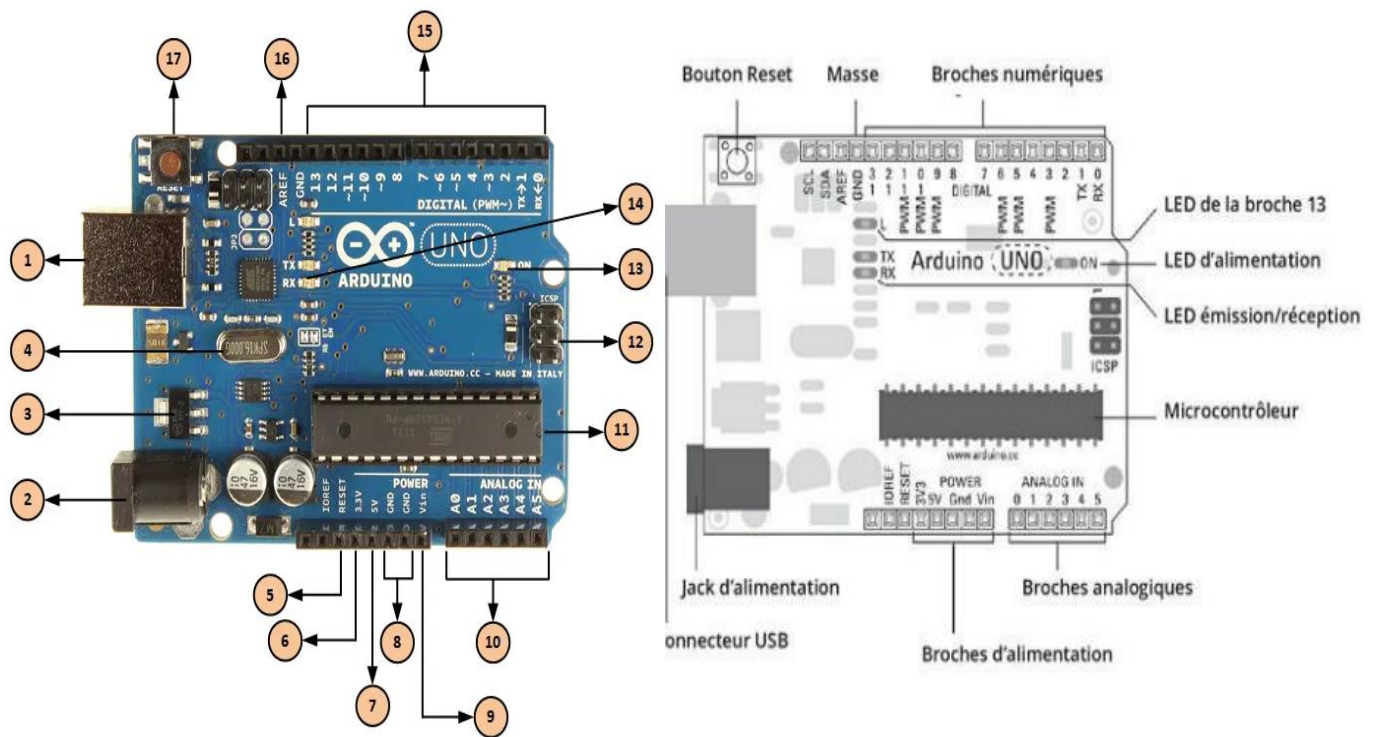


Figure II.9 : Les composants de la carte Arduino UNO.

La majorité des cartes Arduino ont **ces composants** en communs :

1. Alimentation / Programmation par USB : La carte Arduino peut être alimentée par un câble USB connecté à l'ordinateur. Il vous suffit de connecter la carte Arduino à l'ordinateur à l'aide d'un câble de type USB A/B.

2. Alimentation via connecteur Jack DC : La carte Arduino peut être alimentée directement par ce connecteur Jack DC. Ce connecteur est relié au régulateur de tension intégré à la carte. L'alimentation par ce connecteur doit être comprise entre 5 et 12 V.

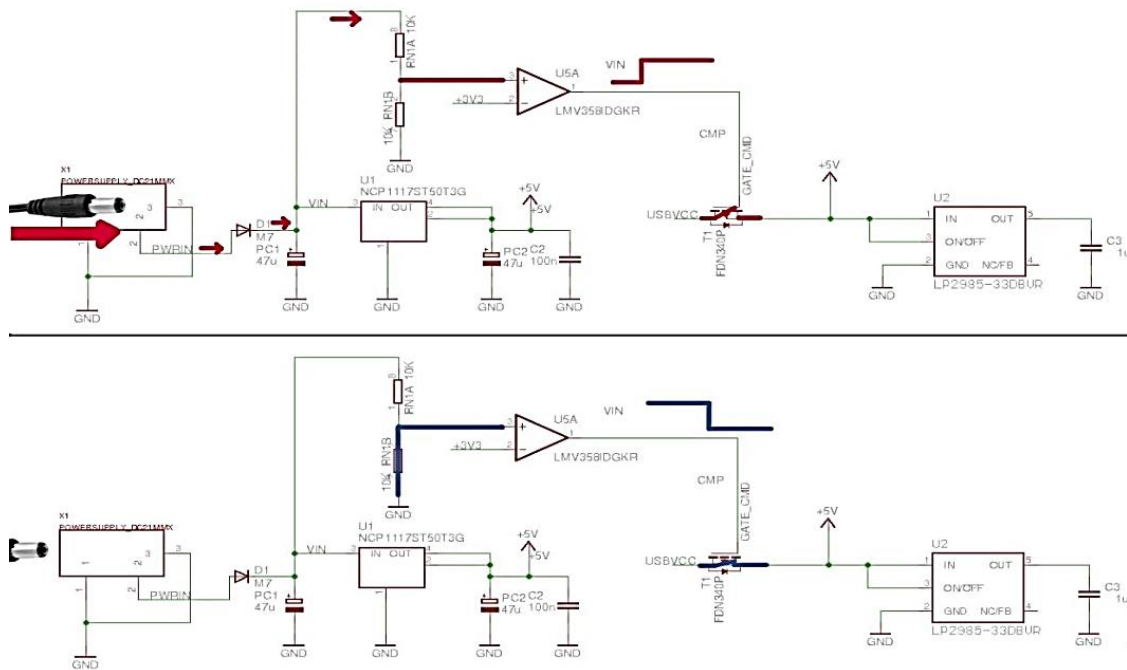


Figure II.10 : Switch entre alimentation avec USB et Adaptateur DC.

3. Régulateur de tension : La fonction du régulateur de tension est de contrôler la tension d'alimentation de l'Arduino pour la stabiliser à la tension correcte du microcontrôleur et de chaque composant de la carte. La tension stable sur la carte Arduino UNO est de 5 volts.

4. Oscillateur à quartz : Un oscillateur à quartz est un composant électronique avec du quartz à l'intérieur, qui vibre sous l'effet piézoélectrique. Les propriétés électromécaniques du quartz permettent au quartz de vibrer à une fréquence très précise. Cet élément aide Arduino UNO à calculer les données temporelles. En haut du composant, vous pouvez lire 16 000. Cela signifie que la fréquence est de 16 000 000 Hz ou 16 MHz.

- 5. Arduino Reset :** En connectant un bouton externe sur la broche de la carte Arduino mentionnée "RESET" il permet de redémarrer votre programme depuis le début.
- 6. 3.3V :** Broche d'alimentation de tension 3.3 Volts.
- 7. 5V :** Broche d'alimentation de tension 5 Volts.
- 8. GND :** Il y a plusieurs broches de ce type présentes sur la carte Arduino, elles sont toutes communes et peuvent être utilisées comme masse (potentiel 0 Volts) pour vos circuits.
- 9. VIN :** Cette broche permet d'alimenter l'Arduino depuis une source de tension extérieure. Elle est reliée au circuit d'alimentation principale de la carte Arduino.
- 10. Broches analogiques :** Arduino UNO dispose de 6 broches d'entrée analogiques, numérotées de A0 à A5. Ces broches sont utilisées pour lire les signaux analogiques des capteurs (tels que les capteurs d'humidité ou de température). La carte Arduino utilise un convertisseur analogique-numérique (convertisseur CAN) pour permettre au microcontrôleur de lire le signal. Un signal sera converti en 10 bits. La valeur peut être lue sur une échelle de 1024 points (de 0 à 1023).
- 11. Microcontrôleur ATmega328p :** Chaque carte Arduino possède son propre microcontrôleur. Vous pouvez le considérer comme le cerveau de l'Arduino. Le microcontrôleur de l'Arduino (figure 11) varie d'une carte à l'autre. Le microcontrôleur provient généralement d'ATMEL. Avant de charger un nouveau programme depuis l'IDE Arduino, nous devons savoir quel est le microcontrôleur de notre carte de développement. Pour la carte Arduino UNO, le microcontrôleur dispose de 28 broches, d'une mémoire flash (32 Ko), d'une mémoire RAM (2 Ko) et d'une mémoire EEPROM (1 Ko).



Figure II.11 : Un microcontrôleur ATmega328P.

- 12. Connecteur ICSP :** Avant tout, le connecteur ICSP (In-Circuit Serial Programming) est une connectique AVR comprenant les broches MOSI, MISO, SCK, RESET, VCC et GND. Il s'agit d'un connecteur de programmation. Ce connecteur permet entre autres de programmer directement le

microcontrôleur sur les couches les plus basses (bootloader, code ASM...). C'est aussi un port appelé port SPI (Serial Peripheral Interface), qui permet de dialoguer avec d'autres composants SPI (écrans, capteurs, etc...).

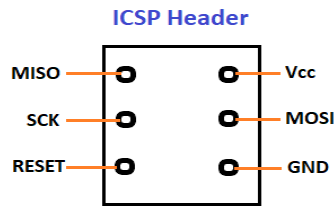


Figure II.12 : Connecteur ICSP.

13. Indicateur LED d'alimentation : Ce voyant doit s'allumer lorsqu'on branche notre carte Arduino sur une source d'alimentation pour indiquer que notre carte est correctement alimentée. Si cette lumière ne s'allume pas, il y a un problème avec notre alimentation.

14. LED RX et TX de Communication : Sur notre carte, nous trouverons deux indicateurs : TX (émission) et RX (réception). Ils apparaissent à deux endroits sur la carte Arduino UNO. Tout d'abord, sur les broches numériques 0 et 1, pour indiquer les broches responsables de la communication série. Deuxièmement, les LEDs TX et RX. Le voyant TX clignote à une vitesse variable lors de l'envoi des données série. La vitesse de clignotement dépend de la vitesse de transmission utilisée par la carte. RX clignote pendant le processus de réception. La vitesse de transmission s'exprime en bauds, soit l'équivalent du bits/seconde si le signal est binaire.

15. Entrées/sorties numériques : Chacune des 14 broches numériques de la Uno peut être utilisée en entrée (input) ou en sortie (output), en utilisant les fonctions *pinMode()*, *digitalWrite()*, et *digitalRead()*. Elles fonctionnent en logique TTL (0V-5V) ; chacune pouvant fournir (source) ou recevoir un courant maximal de 40 mA et dispose si besoin est d'une résistance interne de 'pull-up'.

En outre, certaines broches ont des fonctions spécialisées :

- **Serial :** broche 0 (RX) et broche 1 (TX). Permet de recevoir (RX) et de transmettre (TX) des données séries TTL. Ces broches sont raccordées à leurs homologues sur le chip Atmega8U2 spécialisé dans la conversion USB-to-TTL série.

- **Interruptions externes** 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur LOW, sur un front montant ou descendant, ou encore sur le changement de valeur.

- **PWM :** 3, 5, 6, 9, 10, and 11. Output 8-bit de PWM avec la fonction *analogWrite()*. Les broches étiquetées “~” peuvent être utilisées pour générer des PWM qui est une technique permettant d'obtenir des résultats analogiques avec des moyens numériques.

- **SPI** : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches fournissent le support de communication SPI en utilisant la ‘Library’ spécialisée

- **LED** : 13. Il y a une LED connectée à la broche numérique 13.

La carte Uno a 6 broches d’entrée analogiques, A0 à A5, chacune avec 10 bits de résolution (1024 valeurs différentes).

Par défaut les mesures sont effectuées de la masse à 5V (valeur de référence), mais il est possible de spécifier la valeur de référence en utilisant la broche VREF et la fonction *analogReference()*.

En outre, certaines broches ont des fonctions spécialisées :

- **I2C** : 4 (SDA) and 5 (SCL). Permettent le support du bus I2C (TWI) en utilisant le ‘Library’ Wire.

Autres broches sur la carte :

- **AREF**. Tension de référence déjà mentionnée.

- **Reset**. Permet au niveau bas (LOW° de faire un reset du contrôleur. Elle est utilisée typiquement pour monter un bouton ‘reset’ aux cartes additionnelles (‘Shields’) bloquant celui de la carte principale.

16. Broche AREF : AREF est l’acronyme anglais de “référence analogique”. Cette broche est parfois utilisée pour définir une tension de référence externe (entre 0 et 5 Volts) comme limite supérieure pour les broches d’entrée analogiques.

17. Arduino Reset : Vous pouvez redémarrer un Arduino en appuyant sur le bouton “Reset”. Cela aura pour effet de redémarrer votre programme depuis le début. [13]

Schéma structurel de la connexion Série et alimentation USB :

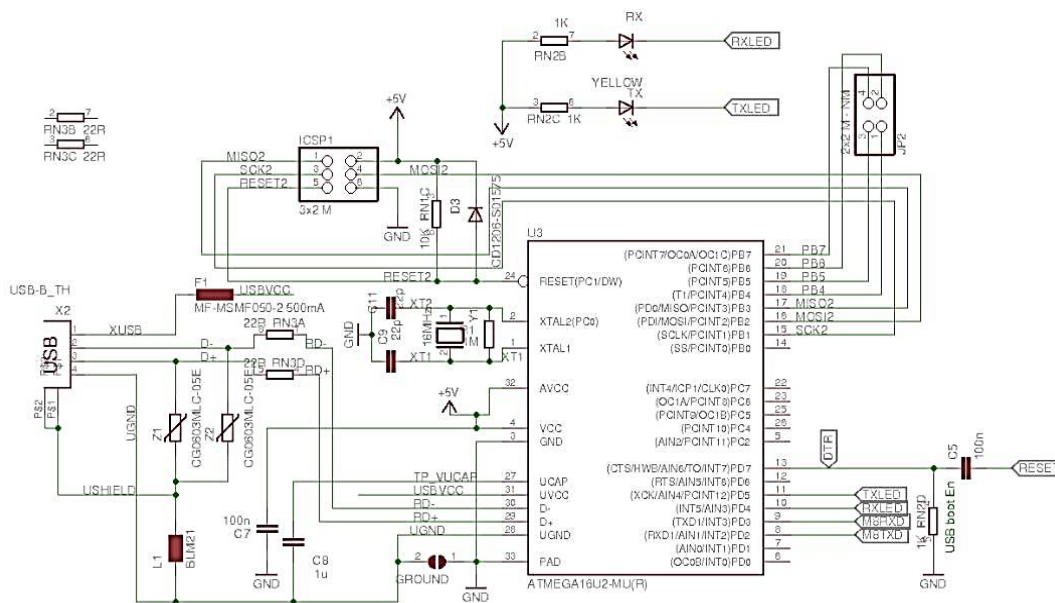


Figure II.13 : USB série/alimentation Bridge

II.2.7.2 Partie logicielle :

- **Le logiciel Arduino IDE :**

Le logiciel Arduino est un environnement de développement gratuit (IDE) open source, qui peut être téléchargé à partir du site officiel d'Arduino.

L'IDE Arduino permet :

- D'éditer un programme : des croquis avec langage proche du C.
- De compiler ce programme dans le langage « machine » de l'Arduino.
- De téléverser (transférer) le programme dans la mémoire de l'Arduino via la liaison USB.
- De communiquer avec la carte Arduino grâce au terminal. [14]

La figure II.14 présente l'interface de programmation Arduino.

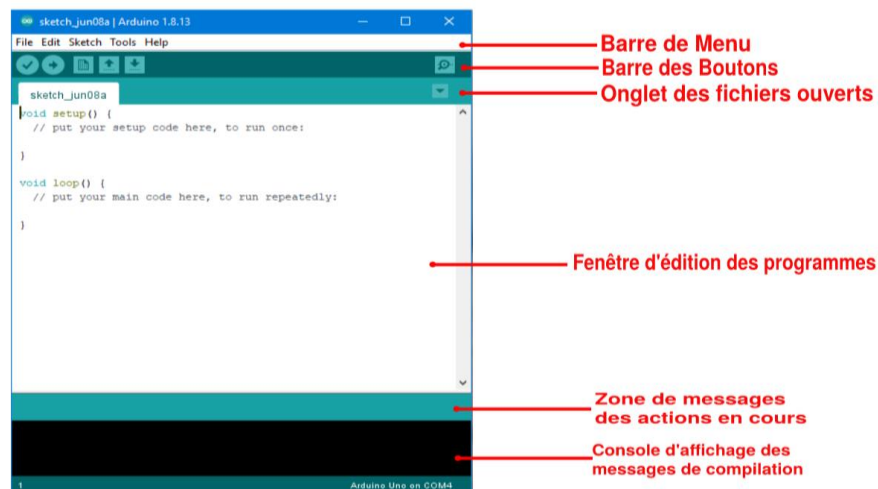


Figure II.14 : Plateforme de l'IDE Arduino.

La « Barre de Boutons » qui donne un accès direct aux fonctions essentielles du logiciel est représentée dans la figure II.15.

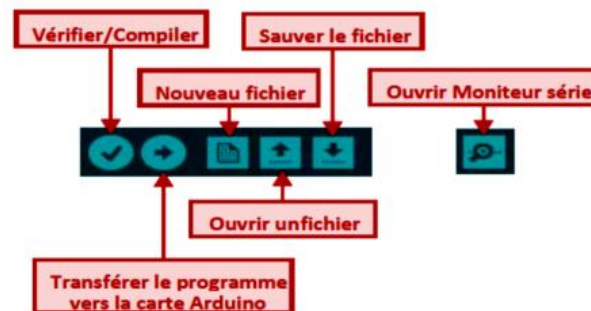


Figure II.15 : Barre de boutons.

- **Le déroulement du programme :**

Le programme se déroule de la façon suivante :

1. Prise en compte des instructions de la partie déclarative
2. Exécution de la partie configuration ‘fonction **setup ()**’
3. Exécution de la boucle sans fin ‘fonction **loop ()**’ : le code compris dans la boucle sans fin est exécuté indéfiniment.

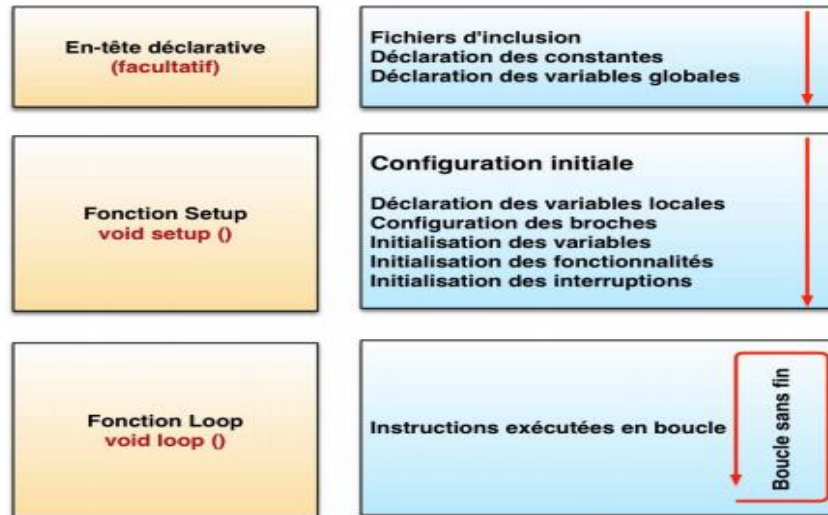


Figure II.16 : Le déroulement du programme.

Si le programme n'est pas compilé vous recevrez un message d'erreur (voir la Figure II.17). [12]

La capture d'écran montre l'interface de l'IDE Arduino 1.0.5-r2. Le code source est visible dans l'éditeur de texte. Le code est le suivant :

```

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltag
  delay(1000); // wait for a second
  digitalWrite(led, LOW); // turn the LED off by making the vol
  delay(1000); // wait for a second
}

```

Une erreur de compilation est affichée dans la console à la ligne 23 : 'expected ')' at end of input'. Cette erreur est soulignée par un cercle rouge.

Figure II.17 : IDE erreur.

II.3 La technologie RFID :

La technologie RFID est basée sur le transfert d'énergie électromagnétique. Pour y parvenir, il doit disposer d'un tag (carte ou puce RFID, également appelé transpondeur) et d'un lecteur RFID.

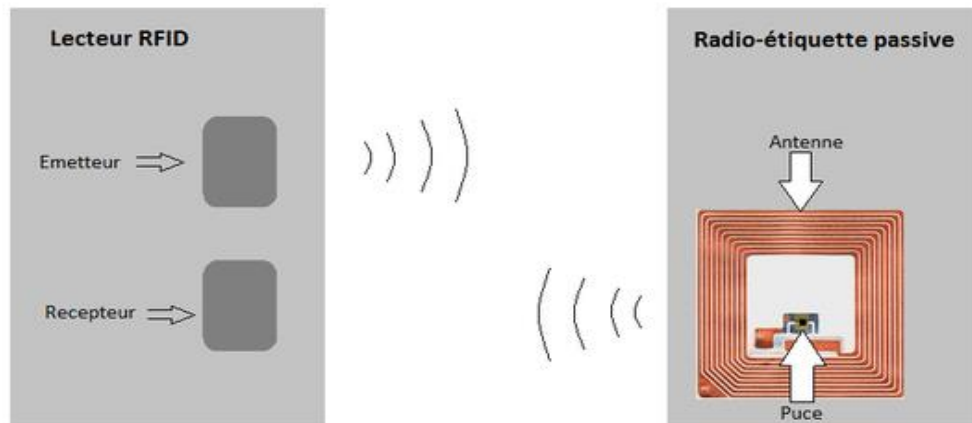


Figure II.18 : La technologie RFID

Quelles applications pour la technologie RFID ?

La RFID est une technologie répandue parmi l'abondance entier. Un des moralités les encore évidents se situe pour les espaces de la logistique et de la saisie : les produits équipés d'une puce RFID peuvent convenir tracés de l'entrepôt à l'administration (suivi fabricant en chaîne de montage, inventaires...). Un jour la fertilité en rayon, la puce RFID devient à la fois un affecté et une maturité d'identification en écran. La RFID est quand longuement utilisée pendant lequel les bibliothèques comme la traçabilité et la toiture des livres.

Mais ce ne sont pas les seules mœurs qui sont faits de la technologie RFID. Parmi les plus connus, on peut avancer :

- Le contrôle d'accès
- Les cartes de paiement sans contact
- Les passeports
- Les puces sous-cutanées des animaux domestiques
- L'accès aux transports publics. [15]

II.3.1 Les cartes RFID :

Les étiquettes RFID, également appelées badge RFID, étiquettes radio ou étiquettes RFID, se composent de deux composants fournis : une antenne et une puce. C'est une puce RFID qui contient

des informations d'identification qui peuvent être lues par un lecteur RFID. Le terme « puce RFID » est parfois utilisé comme synonyme de « tag RFID ».

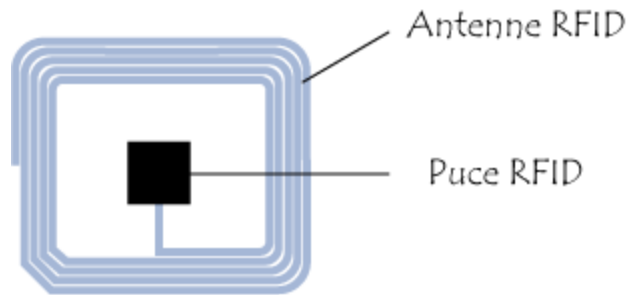


Figure II.19 : Structure d'un Tag RFID

II.3.1.1 Le Fonctionnement de carte RFID :

Les puces RFID (Radio Frequency Identification) sont en silicium (SiO_2). La puce RFID est très fine et inférieure à quelques millimètres. Il a une masse légèrement plus petite, pour lire la puce RFID il suffit de mettre la carte RFID à l'approximation de lecteur RFID, le lecteur doit signaler à l'étiquette qu'elle réagit et retransmettre un signal RF pour lire les informations stocker à la puce. Par exemple, vous pouvez identifier les puces RFID les plus simples et rechercher ou traiter des bases de données pour le contrôle d'accès, le suivi des produits et la gestion des stocks.

II.3.1.2 Les types des Puces RFID :

Il y a trois types de puces RFID :

- La puce RFID passive fonctionne sans batterie. Il est activé par une onde émise par le lecteur RFID.
- La puce RFID active possède une batterie et les informations qu'elle contient sont automatiquement transmises au lecteur RFID.
- La puce RFID intelligente dispose d'un système de sécurité qui peut crypter les informations que vous détenez (par exemple, carte de paiement).

Ces différents types de puces RFID peuvent être écrasés. Cela signifie que les lecteurs RFID peuvent modifier les données pour un usage interne ou personnel. C'est "lecture seule" qui permet aux lecteurs d'apporter des modifications sans changer les données. Accédez à vos données. Le coût de la puce RFID est relativement faible.

II.3.1.3 A quoi servent les puces RFID ?

La technologie RFID a de nombreuses utilisations (figure II.20), y compris les systèmes antivols RFID, les cartes de paiement, les cartes de transfert, les passeports électroniques, les contrôles d'accès et les clés sans contact. Dans l'usine. Protégé contre le vol en même temps que le point de vente. Il est également utilisé pour empêcher la contrefaçon. De nombreuses autres applications sont en cours de développement ou en cours d'utilisation. [16]



Figure II.20 : Les utilisations de la technologie RFID.

II.3.2 Le lecteur RFID.

Le lecteur RFID émet un signal radiofréquence pour activer la puce RFID passant à proximité. À cette fin, il leur a envoyé l'énergie nécessaire à une courte distance. En fonction des performances requises et du type d'application, plusieurs fréquences sont utilisées. Une fréquence plus élevée (exprimée en Hz) permet d'échanger plus d'informations (figure II.21) à un débit plus élevé, tandis qu'une fréquence plus basse présente l'avantage d'une meilleure pénétration.

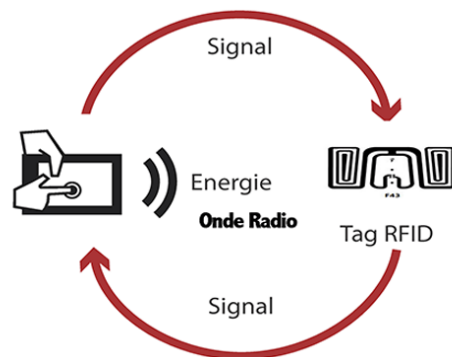


Figure II.21 : La connexion entre le lecteur et la puce RFID.

II.3.2.1 Lecteur RFID : comment ça fonctionne ?

Le lecteur RFID requis pour utiliser la technologie RFID utilise des ondes radio pour envoyer de l'énergie à l'étiquette RFID. C'est grâce à l'antenne émettrice placée simultanément sur le lecteur et l'étiquette. Lorsque l'étiquette RFID est dans le champ magnétique du lecteur RFID, l'appareil demande des informations, reçoit des commentaires et les envoie à chaque application. Il y a deux types de lecteur :

- Le lecteur RFID fixe : sous forme de portique ou de borne, c'est l'objet qui déplace à proximité de lecteur.
- Le lecteur RFID portable : le lecteur qui se déplace à proximité de l'objet pour scanner.

II.3.2.2 Choisir un lecteur RFID : fréquence et portée de lecture

Les lecteurs RFID fonctionnent à plusieurs fréquences : basse (BF 125KHz), haute (HF 13,56 MHz), ultra haute (UHF 860/960 MHz, selon la région du monde), super ultra haute (SHF 2,45 GHz)). (Figure II.22)

Le lecteur de proximité a une courte portée entre 10 et 25 cm, Le lecteur de voisinage a une distance jusqu'à 1m, Le lecteur à moyenne distance entre 1 et 9 mètres et Le lecteur longue portée à plusieurs centaines de mètres.[17]

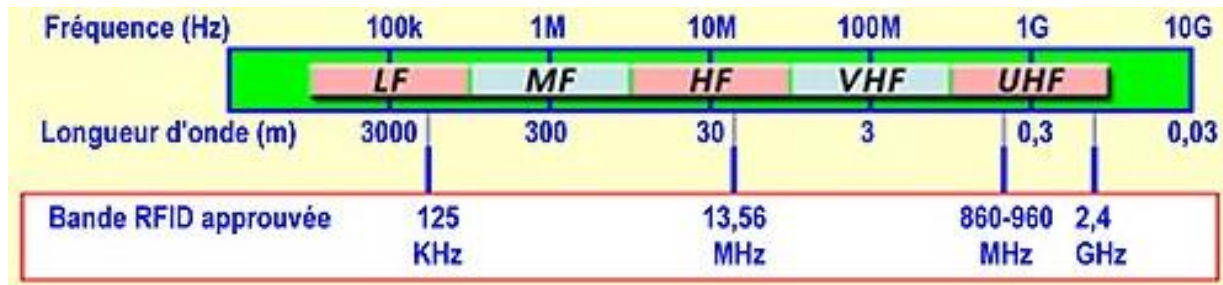


Figure II.22 : Les fréquences et les portées de lecture.

II.3.2.3 Le module RFID RC522 à 13.56MHz :

Ce module est un lecteur de puce RFID à faible coût basé sur le circuit MFRC522, qui est facile à utiliser. Il peut être utilisé dans une large gamme d'applications. MFRC522 est un circuit intégré de lecture/écriture hautement intégré pour la communication sans contact à 13,56 MHz, Portée de lecture : Entre 1 à 3 cm avec la carte et le porte-clés fournis.



Figure II.23 : Le module RFID RC522.

Les caractéristiques du module :

- Basé sur le circuit MFRC522
- Fréquence de fonctionnement : 13,56 MHz
- Tension d'alimentation : 3.3V
- Courant : 13-26mA
- Interface de communication : SPI
- Taux de transfert de données maximum : 10 Mbit / s
- Dimensions : 60mm × 39mm. [18]

II.4 Le module GSM/GPRS :

II.4.1 SIM900 (GSM/ GPRS) Shield :

GPRS / GSM Shield nous offre un moyen de recevoir des données à partir d'emplacements distants en utilisant le réseau GSM. Ce module nous permet d'y parvenir par l'une des trois méthodes suivantes :

- Service de messages courts (SMS)
- Audio

• Service GPRS Le Shield GPRS est compatible avec toutes les cartes avec le même facteur de forme (et disposition des broches) que la carte Arduino standard. GPRS Shield est configuré et contrôlé par son UART à l'aide de simples commandes AT. Basé sur le module SIM900 (figure II.24) de SIMCOM, GPRS Shield est comme un téléphone mobile.

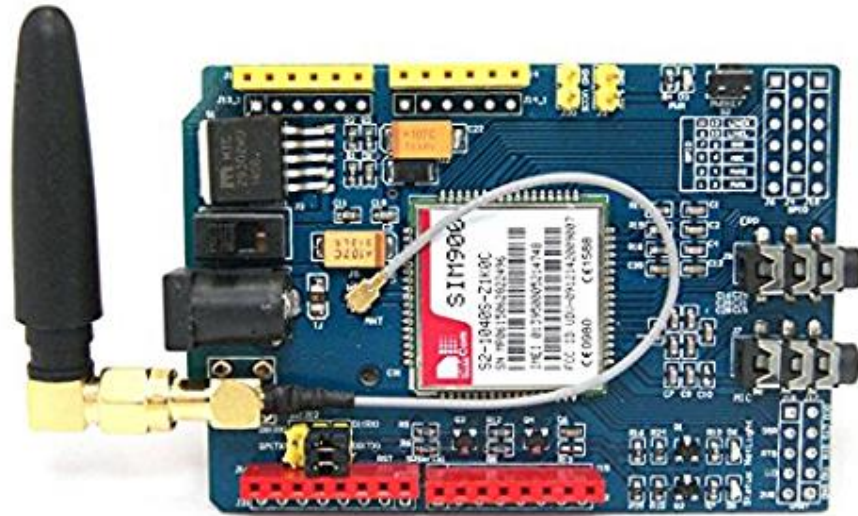


Figure II.24 : SIM900 Shield (GSM/GPRS)

Les caractéristiques :

- Quadri-bande 850/900/1800/1900 MHz-applicable aux réseaux GSM dans tous les pays / régions du monde.
- GPRS multi-slots de classe 10/8
- Station mobile GPRS de classe B
- Conforme à la phase GSM 2/2 +
- Classe 4 (2W à 850 / 900MHz)
- Classe 1 (1W à 1800 / 1900MHz)
- Commande de commande (GSM) 07.07, 07.05 et commande AT améliorée SIMCOM)
- Service de messages courts
- Choisissez le port série libre
- Prise en charge du RTPC avec super condensateur
- Fonction d'activation / désactivation et de réinitialisation prise en charge par l'interface Arduino.

II.4.2 Vue sur la SIM900 :

La figure II.25 représente les composants de la carte SIM900.

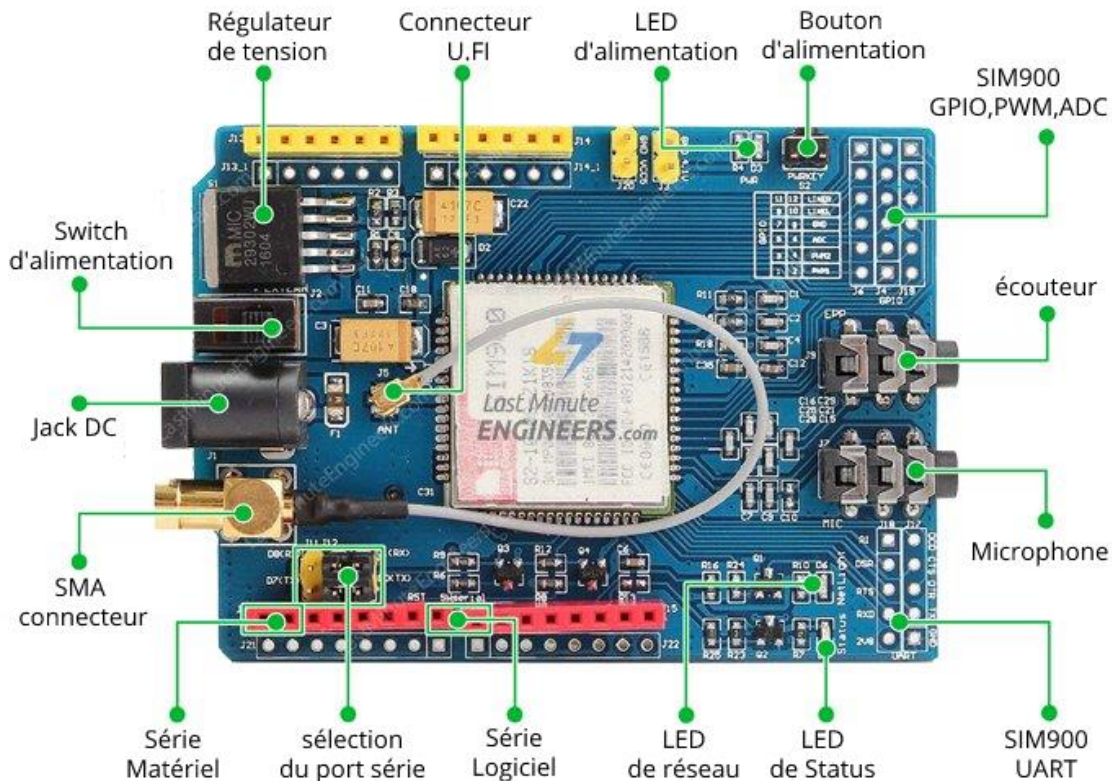


Figure II.25 : les composants de l'extension SIM900.

II.4.3 Les LED indicatrices :

- GSM Shield a trois indicateurs-alimentation du module GSM (Power) : Cette LED est utilisée pour indiquer l'état d'alimentation du capteur GSM.
- SIM900 Power (Status) : Ce voyant est utilisé pour indiquer l'état d'alimentation du SIM900.
- État du réseau (Netlight) : Ce voyant est utilisé pour indiquer l'état du réseau. La LED clignotera lentement ou rapidement selon différents états.

Si nous voulons utiliser des commandes UART, nous devons envoyer AT et placer le cavalier sur la position Hardware. Notre croquis de test utilise une série Software. Par conséquent, nous maintenons généralement des positions Software.

Si vous utilisez UNO. Connectez simplement les broches centrales de SIM900 Shield RX et TX à TX RX sur l'Arduino UNO et modifiez les paramètres du code et donnez le numéro des broches de RX et TX sur Arduino UNO. Dans notre projet, nous associons la carte électronique au module GSM

via le protocole UART. La carte électronique utilisera les commandes AT pour communiquer avec le module GSM et effectuer des transactions de données basées sur la méthode du service de messages courts (SMS).

II.4.4 AT COMMANDS :

L'outil de communication utilisé pour contrôler les modems et les téléphones avec une autre machine est AT COMMANDS (figure II.26). Ce sont des instructions pour exécuter des fonctions dans le système, comme envoyer des messages texte ou passer des appels...

AT : Code d'attention.

C'est le préfixe de la ligne de commande. Il indique au modem qu'une commande ou une séquence de commande est sur le point d'être envoyée.

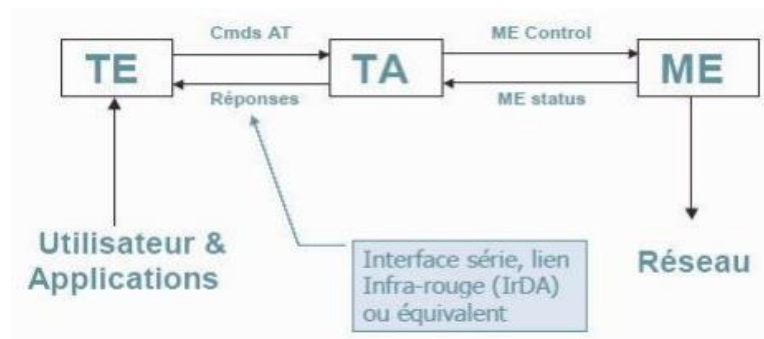


Figure II.26 : Schéma de fonctionnement AT COMMANDS.

La commande ne dépasse pas 556 caractères

"ME" (équipement mobile) : mobile, téléphone portable.

"TE" (Terminal Equipment) : Il peut s'agir d'un ordinateur ou d'un microcontrôleur

"TA" (Terminal Adaptateur) : Fournit un lien entre "ME" et "TE"

« TA » et « ME » forment une seule entité, par exemple, chaque téléphone mobile standard ou un terminal GSM contient « TA » et « ME » dans sa boîte.

TE est une entité indépendante, telle qu'un ordinateur avec un port série, ou un circuit électronique basé sur un microcontrôleur qui implémente le port série. [19]

II.5 Plaque d'essais (Breadboard) :

Cet accessoire s'appelle une breadboard (plaque de test sans soudure). En fait, la plupart d'entre eux sont liés. Ceci est une image rapide pour aider à comprendre.

Les zones rouges et bleu correspondent à l'alimentation (figure II.27). Il y a généralement deux de ces lignes reliant vos composants à la source d'alimentation nécessaire. Par convention, le rouge représente la masse et le bleu représente l'alimentation (+5V, +12V, -5V...). Habituellement, tous les

trous d'une même ligne sont connectés à cette zone. Vous avez donc une ligne électrique qui s'étend tout le long de la carte. Ensuite, vous pouvez voir la zone au milieu. Ces zones sont reliées entre elles par des colonnes. Par conséquent, tous les trous sur la même colonne sont connectés les uns aux autres. Par contre, chaque colonne est différente.

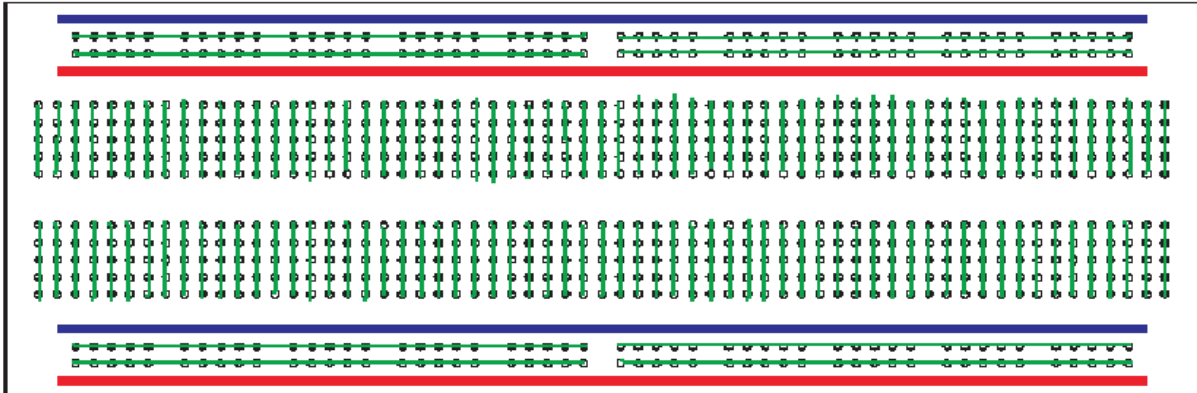


Figure II.27 : Les lignes connecter sur la plaque d'essai.

En superposant des composants sur plusieurs colonnes, vous pouvez les connecter les uns aux autres. Enfin, vous pouvez remarquer qu'un espace divise la carte en deux de manière symétrique. Cet espace coupe également la connexion des piliers. L'espace au milieu est standardisé et doit avoir la largeur d'un circuit intégré standard. En plaçant un circuit intégré au milieu, chaque patte de ce dernier se retrouve sur une colonne, isolée de la précédente et de la suivante. [20]

II.6 Conclusion :

Ce chapitre présente des éléments importants dans la domotique comme le microprocesseur Arduino, lecteur/carte RFID, les capteurs et la puce SIM900. Nous avons défini en détails tous ces composants, qui seront utilisés lors de notre montage. Ce montage est présenté au cours du chapitre 3.

Chapitre III :
**Réalisation d'un prototype de surveillance
d'absences**

III.1 Introduction :

Dans ce chapitre nous allons réaliser notre prototype en utilisant les composants étudiés dans le chapitre précédent, avec la programmation sur le logiciel d'Arduino. Le but de ce système est de pouvoir contrôler et signaler les absences scolaires des élèves par alerte SMS, en munissant chaque élève par une puce RFID, détectable à l'entrée de la salle.

III.2 Le téléchargement et Installation de l'Arduino :

Click sur "Software", (Figure III.1) montre la version (Arduino IDE 1.8.15) disponible pour tous les système (Windows, linux, Mac).



Figure III.1 : Les systèmes d'exploitation qui prise en charge d'Arduino IDE.

Le temps que vous cliquez sur setup, votre navigateur nous amènera à une autre page (Figure III.2). Arduino est un projet open source, il a besoin d'un don pour la continuité des recherches et les développements continu', cliquez sur "Just Download" le téléchargement lancer.

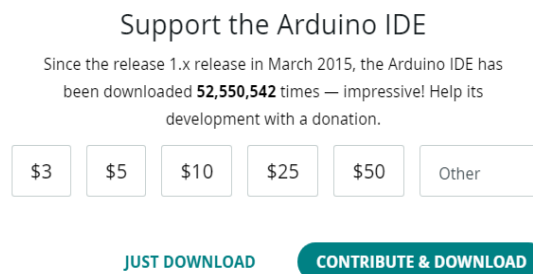


Figure III.2 : La page de don cliquez « Just Download ».

Double clic sur l'application exécutable. La Figure III.3 vous montre le processus en 7 étapes.

Etape 1 : Cliquez sur "I Agree" pour continu.

Etape 2 : Cliquez sur "Next" pour continu.

Etape 3 : Cliquez sur "Install" pour continu.

Etape 4 : Fin d'installation d'IDE.

Etape 5 : Installer Adafruit LLC Ports (COM et LPT).

Etape 6 : Installer Arduino USB Driver Serial.

Etape 7 : Installer Arduino USB Driver LLC.

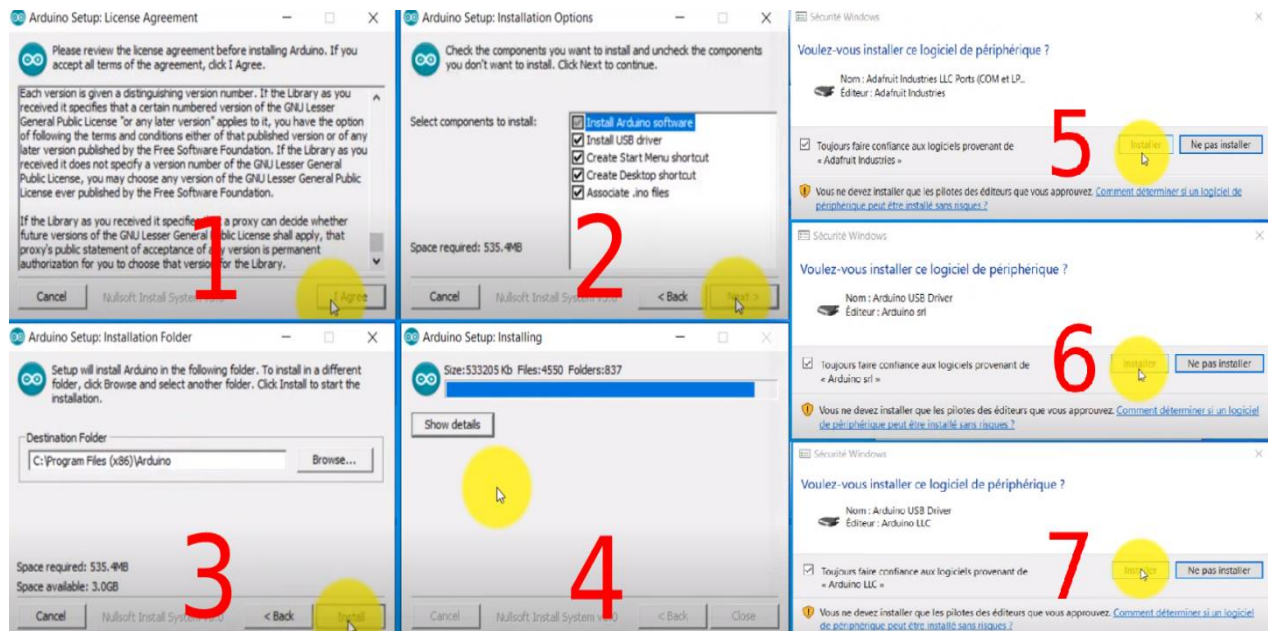


Figure III.3 : Les étapes d'installation d'IDE et les portes COM/serial pour Arduino.

L'application Arduino IDE a été installé avec succès (Figure III.4), et maintenant vous pouvez l'utiliser pour développer votre propre projet, étude, recherche, avec un croquis de langage proche de C.

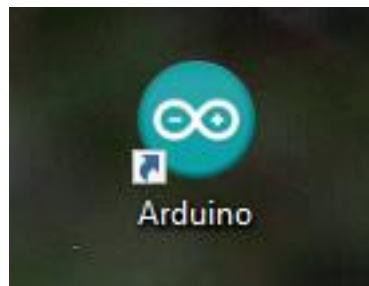


Figure III.4 : L'application IDE été installer (Icon sur le Desktop).

III.3 Schéma générale de câblage :

La figure III.5 représente le schéma général de notre circuit. Ici tout commence avec le lecteur RFID après la lecture des cartes RFID des élèves, puis lui transférer les données au Arduino qui voit les identités et décide s'il est connu ou non, dans le cas d'absence l'Arduino doit envoyer une commande AT de tourner la SIM900 en mode messagerie avec le numéro des parents de ce qu'il est absent, la SIM900 envoyer un SMS aux parents.

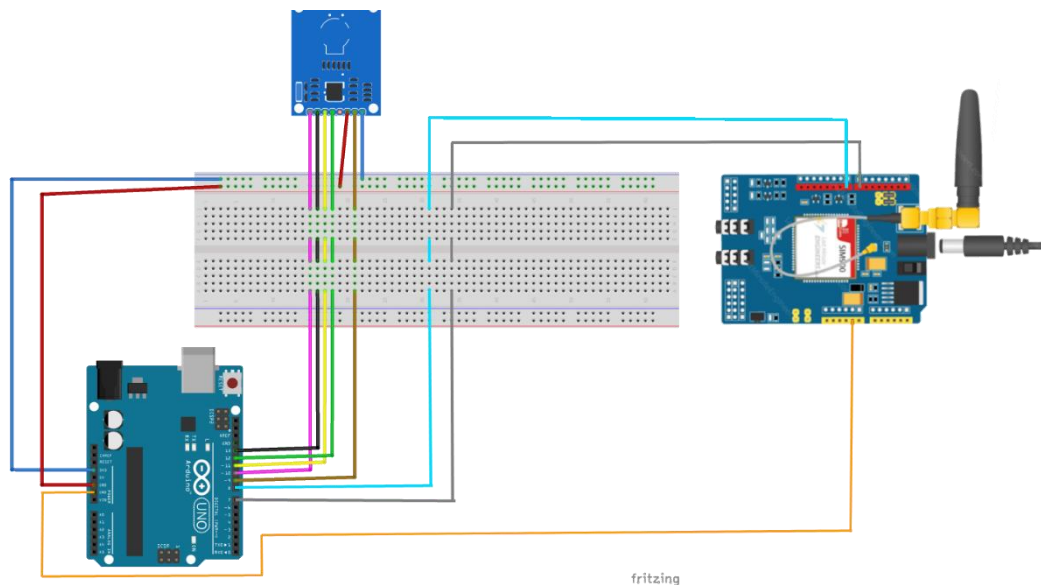


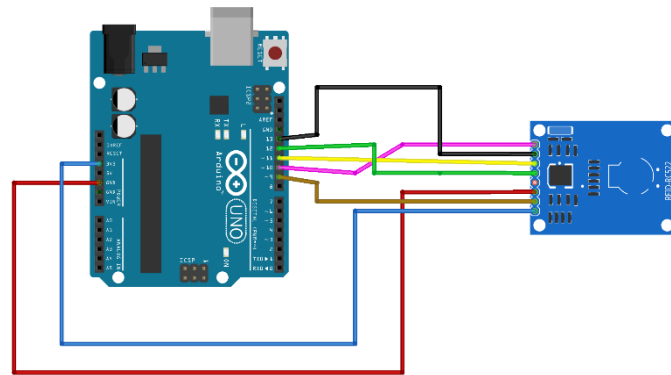
Figure III.5 : Schéma général de circuit.

III.3.1 Câblage Arduino + RFID lecteur :

Le protocole 'SPI' et la bibliothèque 'MFRC522' sont utilisés par le module RFID RC522 pour communiquer avec Arduino. 'SPI' utilise des broches précises sur la carte Arduino (Figure III.6).

Le brochage est comme suit (à gauche côté Module RC522, à droite côté Arduino UNO) :

- Vcc <-> 3V3 (ou 5V selon la version du module).
- RST (Reset) <-> 9.
- GND (Masse) <-> GND.
- MISO (en anglais : Master Input Slave Output) <-> 12.
- MOSI (en anglais : Master Output Slave Input) <-> 11.
- SCK (en anglais : Serial Clock) <-> 13.
- SS/SDA (en anglais : Slave select) <-> 10. [21]



fritzing

Figure III.6 : Le brochage de Arduino avec le module RFID RC522.

III.3.2 Câblage Arduino + SIM900 :

La plaque SIM900 (GPRS/GSM Shield) utilise le protocole 'Software Serial' pour communiquer avec Arduino. Vous pouvez faire un switch entre la communication software et hardware au niveau de la plaque SIM900 placez les **cavaliers** de la carte de manière à utiliser les broches 7 et 8 pour communiquer entre eux (Figure III.7) et n'oubliez pas d'insérer la carte SIM sous la plaque SIM900, alimentation externe de 12V-2A.

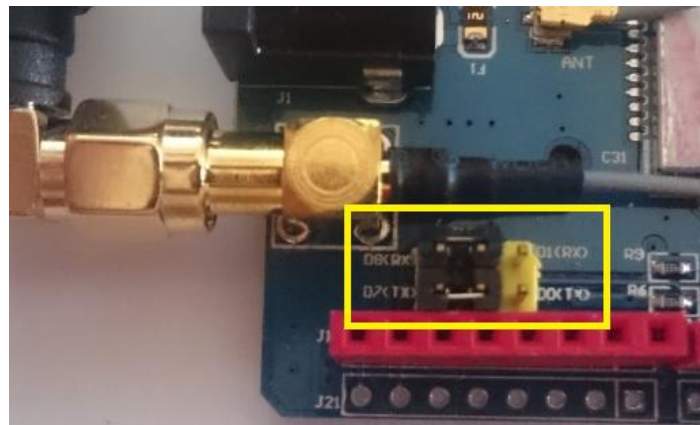


Figure III.7 : Switch les cavaliers au (D8, D7)

Le brochage se fait comme suit (à gauche côté SIM900, à droite côté Arduino UNO) :

- D7 (Tx de la plaque SIM900) <-> PIN7 (Rx de la carte Arduino).
- D8 (Rx de la plaque SIM900) <-> PIN8 (Tx de la carte Arduino).
- GND (Masse) <-> GND.

Le protocole 'Software Serial' utilise des broches spécifiques des microcontrôleurs Arduino, dans la figure suivante (Figure III.8), vous verrez la façon dont nous avons branché l'Arduino a SIM900.

Ici on a branché les PIN7, PIN8, GND de l'Arduino avec D7, D8, GND de la plaque SIM900, Pour activer la communication transmet/recevez entre les deux il faut d'abord initialiser les broches par IDE dans le croquis.

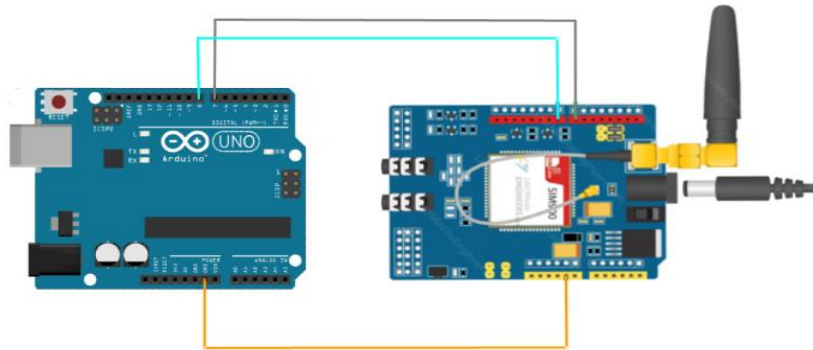


Figure III.8 : Le brochage entre Arduino et SIM900.

La figure III.9 représente le branchement réel réalisé au cours de ce projet.

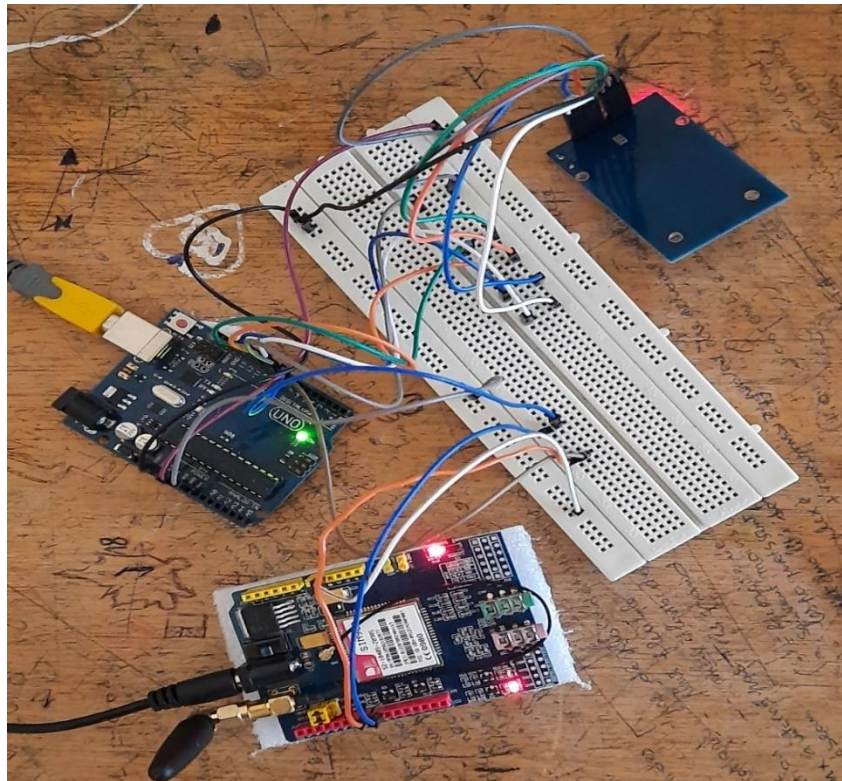


Figure III.9 : Le schéma réel du circuit

III.3.3 Le fonctionnement de circuit au niveau de Arduino IDE :

Etape 1 :

Il faut que l'élève mette sa carte à proximité de lecteur RFID (proximité de 1 à 2cm) pour que le lecteur RFID lise la carte et décide si cet ID appartient à cette classe ou non. (Figure III.10)

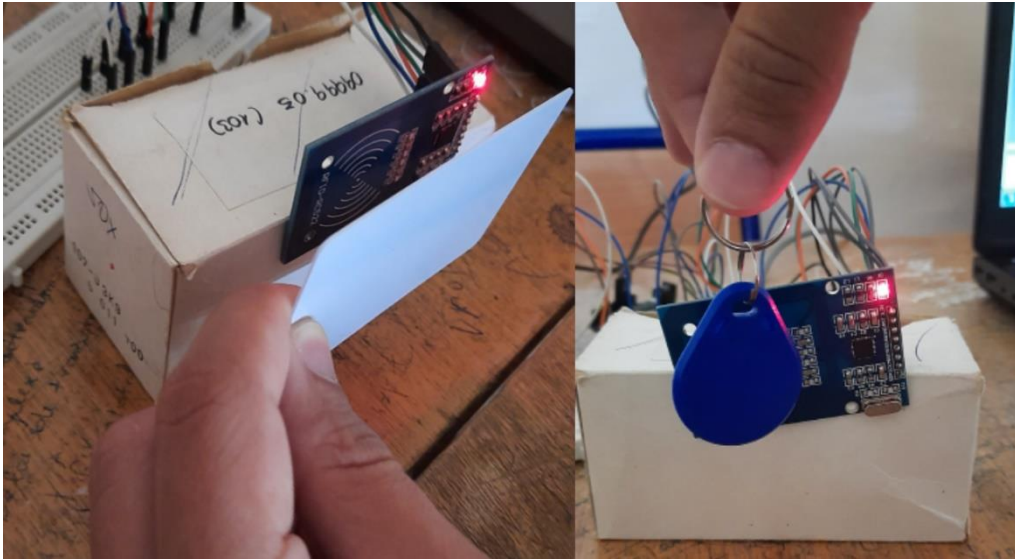


Figure III.10 : L'élève met sa carte à proximité de lecteur RFID.

Etape 2 : Si le lecteur le reconnaît, cela veut dire qu'il est dans cette classe, le serial monitor va afficher le ID de la carte et le nom de l'élève, s'il ne le reconnaît pas, cela veut dire qu'il n'est pas de cette classe, le serial monitor va afficher un message « utilisateur non inscrit ». (Figure III.11)

```
COM4
ID de la Carte : 215 32 66 115
eleveA
utilisateur non inscrit
```

Figure III.11 : Les deux cas de lecteur RFID (connait/ne reconnaît pas) la carte RFID.

III.3.4 Le fonctionnement de circuit au niveau l'interface graphique :

III.3.4.1 Construire l'interface :

Pour l'industrie de l'interface utilisateur, nous avons utilisé le programme Visual basic 6.0, Parce que cette version est gratuite, et il n'y a aucun danger qu'elle soit utilisée, surtout lors d'un événement officiel, comme le master ou le doctorat.

- **L'interface de Système :**

Tout d'abord, nous allons dans l'application et de nous montrer une fenêtre de **New project** on choisit **Standard Exe** puis on clique sur **Open**. (Figure III.12)

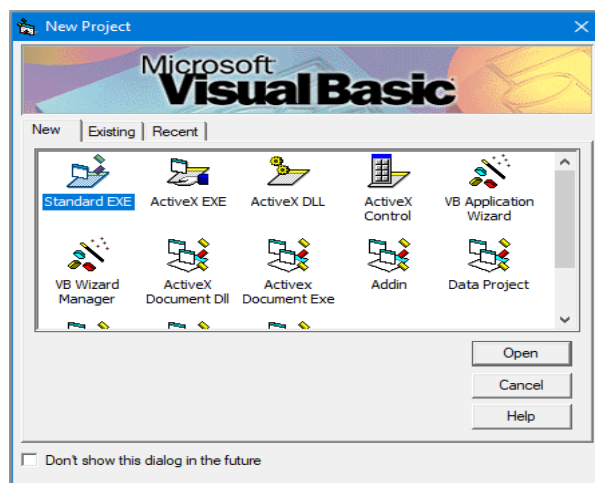


Figure III.12 : Fenêtre de nouveau projet.

Puis une nouvelle fenêtre principale apparaît, au milieu il y a une autre fenêtre c'est la **Forme** de notre interface. (Figure III.13)

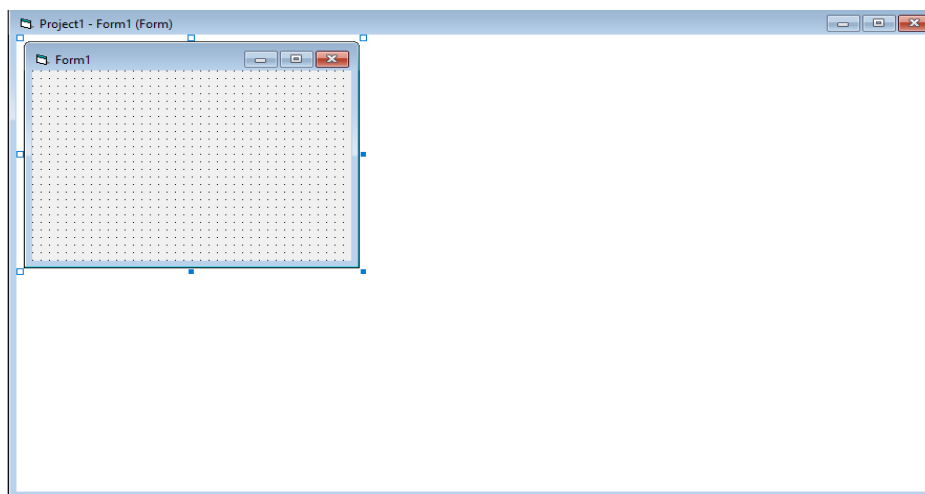


Figure III.13 : Fenêtre de l'interface au milieu de fenêtre de projet.

Dans la Figure III.14, nous vous montrerons les outils que nous avons utilisés dans ce menu.

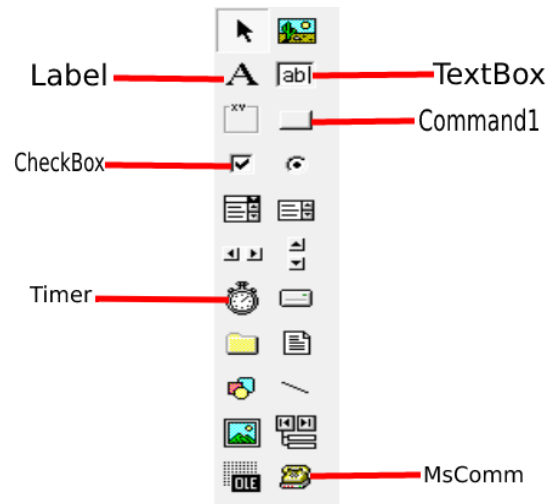


Figure III.14 : Boite d'outils de Visual basic.

On sort huit **Labels**, on les appelle comme sur la Figure III.15.

Label c'est un champ des caractères qui nous aide à comprendre l'interface, on ne peut pas copier ou coller dans le Label.

Label2 : « La date : »

Label6 : « Horloge : »

Label1 : « Temp écoulé : »

Label4 : « Mins et Secs »

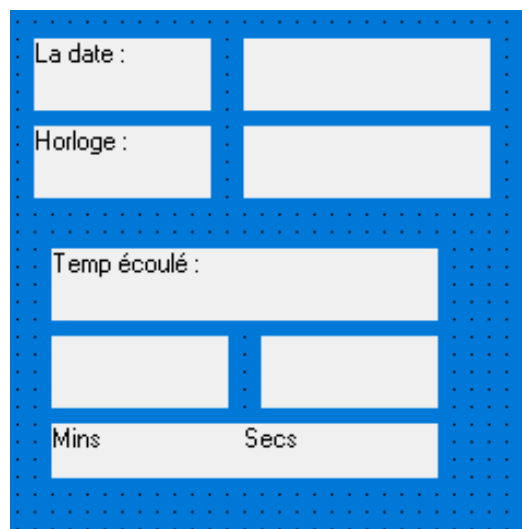


Figure III.15 : Les labels et ses noms.

On sort un **textBox** et posé avec une grande taille (Figure III.16), et deux petites textBox pour le compteur de fenêtre de message pour rappelle l'enseignant d'envoyer les SMS, ces 2 textBox doit être invisible.

TextBox : c'est un champ d'écriture soit d'après l'utilisateur ou d'après le programme lui-même, on peut copier et coller dans ce champ.

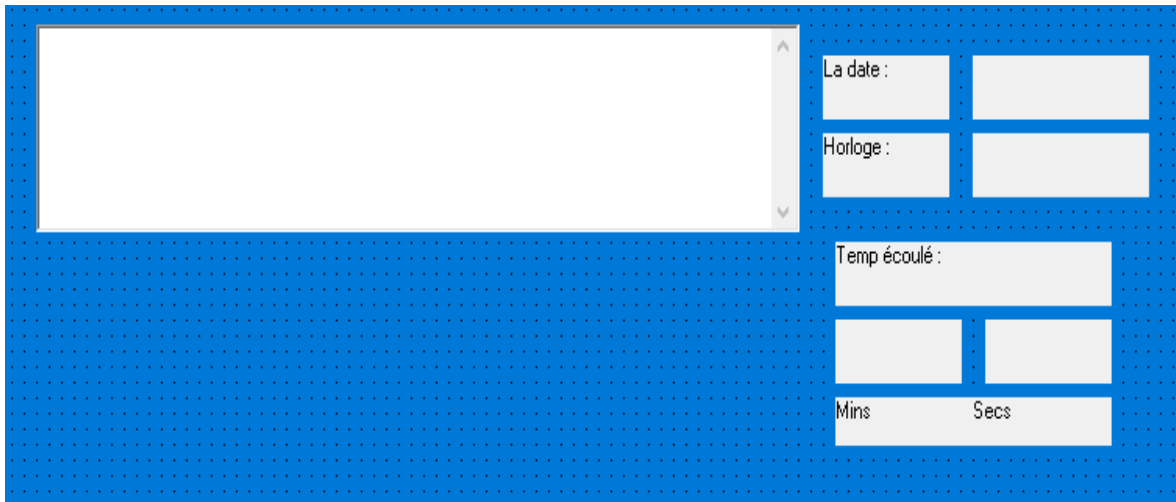


Figure III.16 : Ajouter un grand textBox.

Puis on ajoute quatre **CheckBox**, deux pour capter les noms de **textBox** et on les appelle 1et 2 car ces deux ne soient pas visible en **Form**. Les deux pour envoyer les SMS on les appelle « Elève1, Elève2 », Voir Figure III.17.

CheckBox : c'est une case de coché décoché et prend deux valeur 0 et 1.

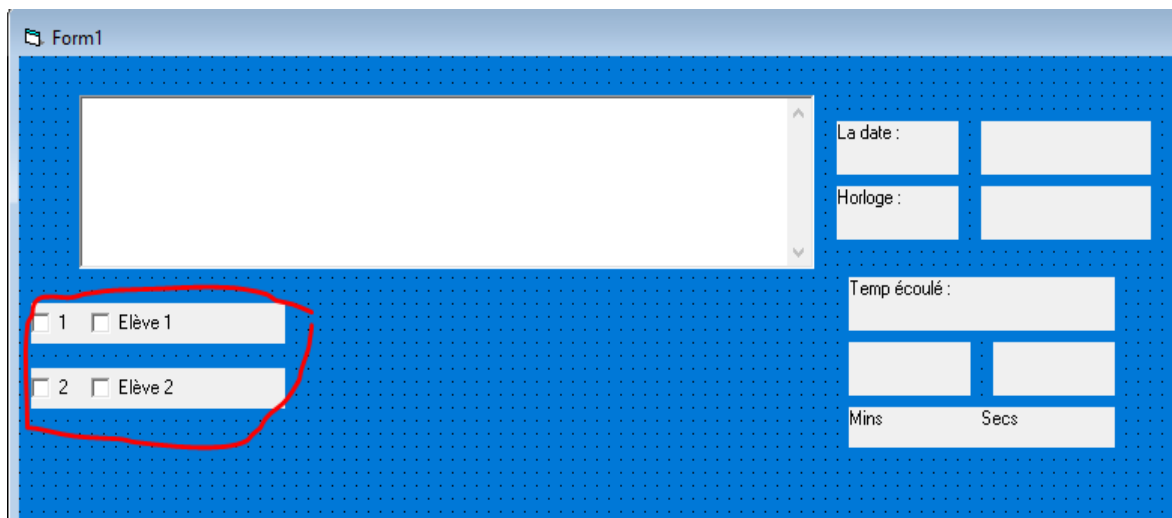


Figure III.17 : l'emplacement des quatre CheckBox.

On ajoute les éléments suivants : (voir Figure III.18)

Bouton **Command1**, et trois **Timers** et **MsComm1** « entouré d'une ligne rouge ».

Command1 : c'est un bouton d'une action appliqué au temps vous appuyez sur le.

Timer : c'est outil de temp vous pouvez utiliser pour soit affiché le temp actuelle ou comme un compteur qui compte les seconds.

MsComm1 : c'est un outil qui crée une porte de communication en série entre l'interface et l'Arduino UNO. (Installation de MsComm en Figure III.19)

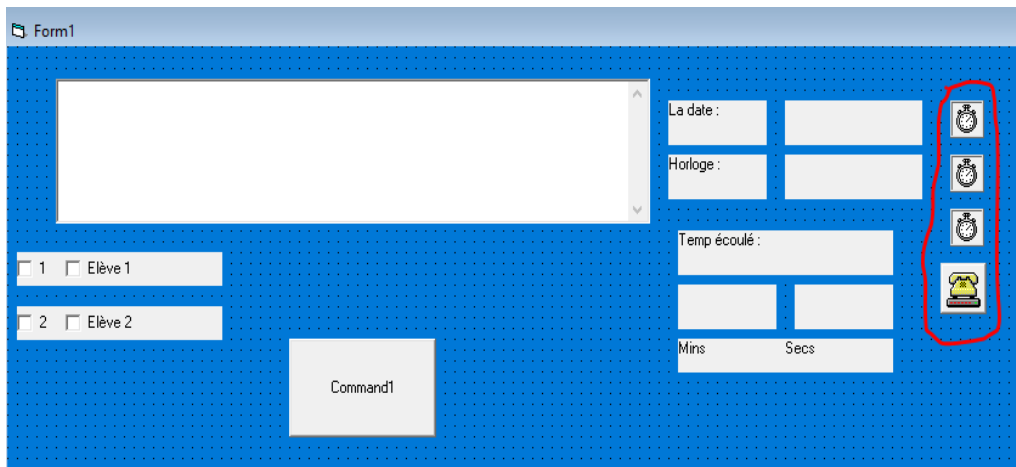


Figure III.18 : La position des trois Timers, un bouton, MsComm1.

Pour installer **MsComm** Cliquez avec le bouton droit de la souris sur la liste d'outils et vous choisissez **Components**, coché sur **Microsoft comm control 6.0 (SP6)** puis **Apply**. (Figure III.19)

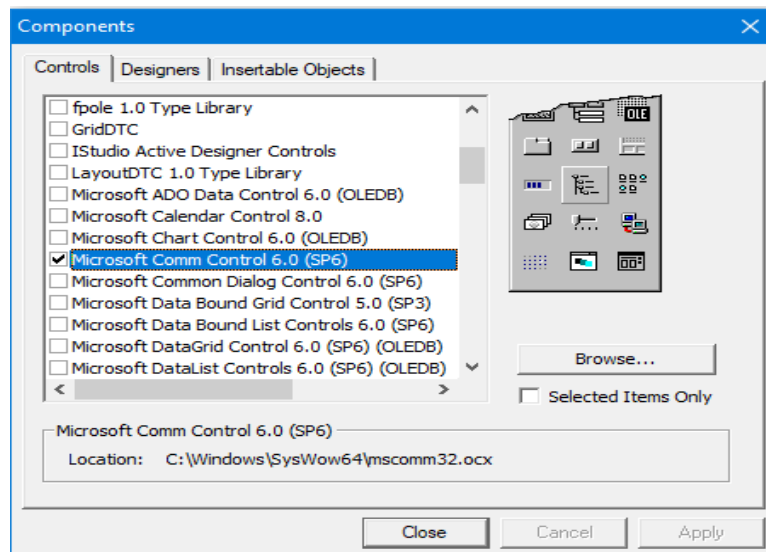


Figure III.19 : Installation de MsComm.

- Le Code de programmation de Système :

Et maintenant on passe à la programmation, double cliquer sur la surface de la **Form** et on écrit le code de programmation afficher sur la Figure III.20.

```
Private Sub Form_Load()
lblDate.Caption = Format(Now(), "short date") ' La date afficher dans l'interface
mins = 0
Timer3.Enabled = True ' timer3 utiliser pour calculer les minutes et les seconds

'MSComml.Settings = "9600,n,8,1"
'MSComml.CommPort = 10
'MSComml.PortOpen = True

With MSComml
    .CommPort = 4
    .Settings = "9600,N,8,1"
    .Handshaking = comRTS
    .RTSEnable = True
    .DTREnable = True
    .RThreshold = 1
    .SThreshold = 1
    .InputMode = comInputModeText
    .InputLen = 0
    .PortOpen = True
End With
End Sub
```

Figure III.20 : Codage de la Form.

Puis double cliquer sur **Timer_1** et on écrit le code suivant : (Figure III.21)

```
Private Sub Timer1_Timer()
txtrec.Text = MSComml.Input
search = txtrec.Text
If (InStr(search, "eleveA")) Then 'lecteur RFID lire ID et le nom d'élève A
Check1.Value = 1

End If
If (InStr(search, "eleveB")) Then 'lecteur RFID lire ID et le nom d'élève B
Check2.Value = 1

End If
End Sub
```

Figure III.21 : Codage de Timer_1.

Et double cliquer sur **Timer_2**, on écrit le code apparent dans la Figure III.22.

```
Private Sub Timer2_Timer()
Label5.Caption = Time ' Horloge (le temp afficher dans l'interface)
End Sub
```

Figure III.22 : Codage de Timer_2.

Puis le **Timer_3**, le code est en Figure III.23.

Ce Timer qui décide qui est absent est continue les recorde de lecture des cartes RFID jusqu'à 2 minutes c'est le temp maximal pour cocher directement les élevés qui sont absents, et avec une fenêtre pour rappeler au professeur d'envoyer les SMS a temp égale à 3 minutes.

```

Private Sub Timer3_Timer() 'calculer les mins et les secs et le tmp pour coché les absents
secs = secs + 1
Label7.Caption = secs ' afficher les seconds dans Label7 a l'interface
If secs > 59 Then
secs = 0
mins = mins + 1
Label3.Caption = mins ' afficher les minutes dans Label3 a l'interface
End If
If ((mins < 1) And (Check1.Value = 1)) Then
Check3.Value = 0
Else
If ((Check1.Value = 0) And (mins > 1)) Then 'si l'Eleve 1 venu apres 2 min , absent
Check3.Value = 1
End If
End If
If ((mins < 1) And (Check2.Value = 1)) Then
Check4.Value = 0
Else
If ((Check2.Value = 0) And (mins > 1)) Then 'si l'Eleve 2 venu apres 2 min , absent
Check4.Value = 1
End If
End If
txtMins.Text = mins
txtSecs.Text = secs
If ((txtMins.Text = "3") And (txtSecs.Text = "1")) Then
MsgBox " C'est le temp envoyer les SMS ", vbExclamation, Title:=" Alert ! "
End If
End Sub

```

Figure III.23 : Codage de Timer_3.

En fin double clique sur le bouton **Command1**, est écrit le code suivant : (Figure III.24)

Dans cette action si vous n'installez pas MsComm vous ne pouvez rien envoyer car il n'y a pas de porte ouverte pour communiqué entre l'interface graphique et l'Arduino.

L'élément MsComm permet de transférer les DATA au microcontrôleur.

```

Private Sub Command1_Click() 'ce Boutton est pour envoyer les SMS
If (Check3.Value = 1) Then
MSComm1.Output = "1" ' Envoyer a l'Arduino : DATA="1" pour SMS au parents d'élèveA
End If
If (Check4.Value = 1) Then
MSComm1.Output = "m" ' Envoyer a l'Arduino : DATA="m" pour SMS au parents d'élèveB
End If
End Sub

' PS: les numéros des parents sont stocké au niveau de croquis d'Arduino

```

Figure III.24 : Codage de bouton Command1.

Les noms des élèves sont juste à l'interface car l'Arduino peut envoyer juste un code qui identifier la carte RFID, donc c'est l'interface qui traduit ce code en nom d'élève.

Pour mieux comprendre, l'Arduino sait juste que le ID : **215 32 66 115**, est associé à **eleveA**, ce dernier peut être un chiffre ou un code ce n'est pas obligé d'écrire le nom exact. (Figure III.25)

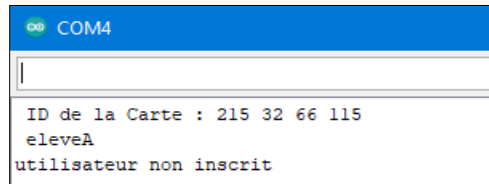


Figure III.25 : Affichage de sérial moniteur.

Donc c'est le programme de l'interface qui contient les noms des élèves mais juste à l'interface car elle traduit le nom venu de l'Arduino en action de cochage après certaines temp (la condition dans l'interface contient le même nom qui est on croquis d'Arduino, l'interface fait un scanne pour voir si ces noms sont afficher dans le grand **textBox**. Exemple : dans l'Arduino : **eleveA**, mais dans Visual basic : **eleveA** dans le code et **Elève 1** a l'interface finale.

Les numéros des parents sont stockés dans le **Croquis** d'Arduino et souligner en rouge (Figure III.26).

Remarque : Chaque ligne est expliquée par un commentaire.

```

////////////////////////////////////
void SendMessage1(String message)
{
  mySerial.println("AT+CMGF=1"); // AT command pour envoyer le SMS
  delay(1000); // pause de 1000 milli seconds ou 1 second

  mySerial.println("AT+CMGS=\"+213xxxxxxxxx\""); // taper ici le Numero des parents de l'Eleve 1
  delay(1000);
  mySerial.println(message); // message pour envoyer
  delay(100);
  mySerial.println((char)26); // terminer AT command avec ^Z, ASCII code 26 (C-A-D: Control + z)
  delay(100);
  mySerial.println();
  delay(5000); // donner le temp au module d'envoyer le SMS
}
////////////////////////////////////
void SendMessage2(String message)
{
  mySerial.println("AT+CMGF=1"); // AT command pour envoyer le SMS
  delay(1000); // pause de 1000 milli seconds ou 1 second

  mySerial.println("AT+CMGS=\"+213xxxxxxxxx\""); // taper ici le Numero des parents de l'Eleve 2
  delay(1000);
  mySerial.println(message); // message pour envoyer
  delay(100);
  mySerial.println((char)26); // terminer AT command avec ^Z, ASCII code 26 (C-A-D: Control + z)
  delay(100);
  mySerial.println();
  delay(5000); // donner le temp au module d'envoyer le SMS
}

```

Figure III.26 : Les numéros des parents dans le Croquis.

III.3.4.2 Travaillé avec l'interface :

Etape 1 : Si l'élève vient après un temp spécifiée par le programmeur, il sera inscrit comme absent.

L'interface GUI dessiné en Visual basic version 6. L'élève qui est absent doit être coché. (Figure III.27)

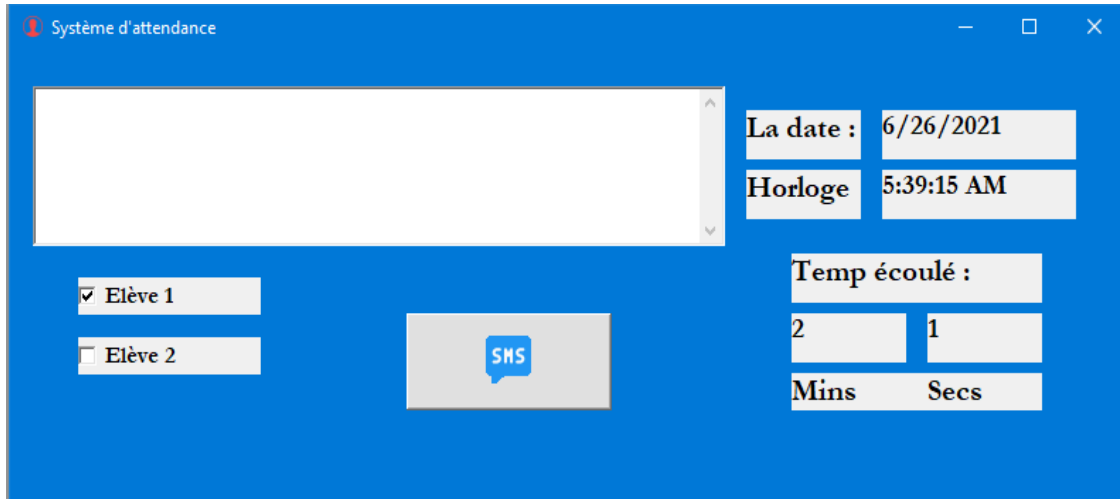


Figure III.27 : Une interface simple programmer pour capte les absences.

Etape 2 :

Une fenêtre de message d'information va afficher à l'enseignant après certain temp pour lui rappeler d'envoyer les messages aux parents des élevés qui sont absents. Avec une option de modification l'enseignant peut ajouter ou supprimer les absences. (Figure III.28)

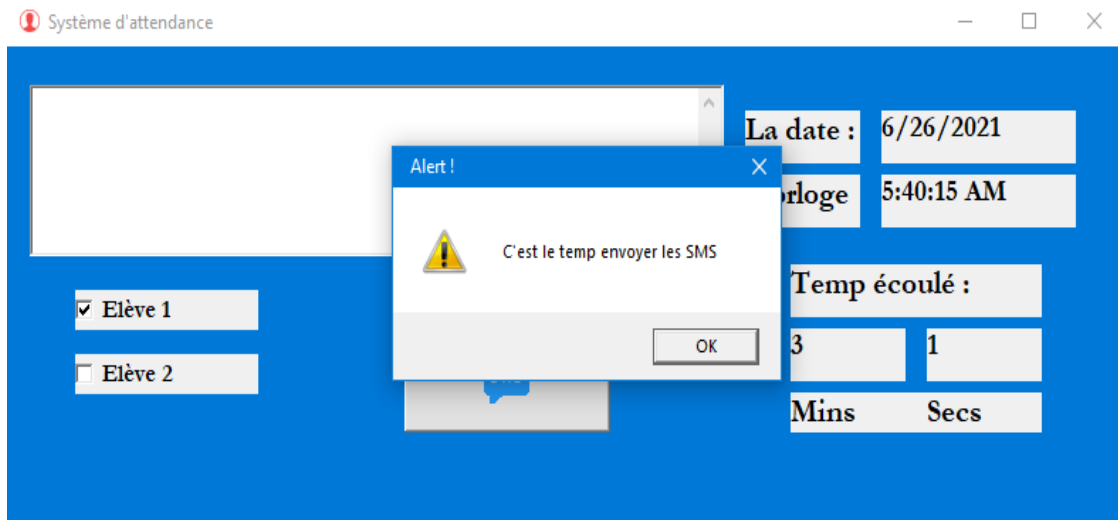


Figure III.28 : Fenêtre de rappel à l'enseignant pour envoyer les SMS.

Etape 3 :

Une fois l'enseignant clique sur envoyer SMS (Botton SMS en bleu), un message d'information sera reçu par les parents. (Voir Figure III.29)



Figure III.29 : Un exemple d'un SMS recevez par les parents.

III.4 Conclusion :

Nous avons pu réaliser le circuit souhaité pour l'alerte des parents par SMS sur les absences scolaires des enfants à l'aide d'un carte et lecteur RFID, une carte SIM900 et surtout un Arduino. Cette application permet de rassurer les parents en direct sur leurs enfants.

Conclusion générale

L'intelligence artificielle nous entoure et fait partie de notre vie quotidienne : à l'école, au travail et même dans la rue, nous utilisons nos téléphones, ordinateurs ou tablettes. Avec le développement de la technologie, les besoins de l'homme augmentent. Son désir d'utiliser la technologie pour tout contrôler s'étend à son environnement. C'est dans cette optique qu'est née la domotique : un ensemble de technologies utilisées dans les bâtiments, visant à centraliser et automatiser certaines tâches et à rendre l'utilisateur capable de communiquer avec divers équipements locaux grâce à la mobilité intégrée à ces systèmes.

Avant tout, les nombreux apports que cette technologie permet ne sont pas des moindres. En répondant aux nouveaux enjeux sociaux, économiques et environnementaux de la société, cet outil peut être très prometteur de par le rôle qu'il joue dans l'amélioration du cadre de vie par l'amélioration du confort, de la communication, de la sécurité et de la gestion de l'énergie.

Et pour cela, dans notre travail nous avons pu mettre sur pied système qui faciliterait la communication entre les différents acteurs du système éducatif. Cette communication doit se faire uniquement à travers des SMS qui seront expédiés à partir du système en fonction des circonstances et des groupes de personnes ciblées.

A travers ce système nous comptons améliorer la communication entre les autorités scolaires et les parents d'élève et tuteurs. Où ce système repose sur le principe d'alerter les parents en cas d'absence ou de retard de leurs enfants dans leurs établissements scolaires, notamment au vu du phénomène généralisé du décrochage scolaire.

Annexes

Annexe 1 : Programme pour lire le ID de chaque Carte RFID.

```
#include <SPI.h>
#include <MFRC522.h>
// INPUT
#define SS_PIN 10
#define RST_PIN 9

MFRC522 rfid(SS_PIN, RST_PIN);
// Init array that will store new NUID
byte nuidPICC[4];
void setup() {
  Serial.begin(9600);
  SPI.begin(); // Init SPI bus
  rfid.PCD_Init(); // Init MFRC522
  Serial.println(F("Scan RFID NUID..."));
}
void loop() {
  readRFID();
  delay(200);
}
void readRFID()
{
  // Look for new card
  if (! rfid.PICC_IsNewCardPresent())
    return;
  // Verify if the NUID has been readed
  if ( !rfid.PICC_ReadCardSerial())
    return;
```

```

if (rfid.uid.uidByte[0] != nuidPICC[0] ||
    rfid.uid.uidByte[1] != nuidPICC[1] ||
    rfid.uid.uidByte[2] != nuidPICC[2] ||
    rfid.uid.uidByte[3] != nuidPICC[3] ) {
    Serial.println(F("A new card has been detected."));
    // Store NUID into nuidPICC array
    for (byte i = 0; i < 4; i++) {
        nuidPICC[i] = rfid.uid.uidByte[i];
    }
    Serial.print(F("RFID tag in dec: "));
    printDec(rfid.uid.uidByte, rfid.uid.size);
    Serial.println();
}
// Halt PICC
rfid.PICC_HaltA();

// Stop encryption on PCD
rfid.PCD_StopCrypto1();
}
/**
 * Helper routine to dump a byte array as dec values to Serial.
 */
void printDec(byte *buffer, byte bufferSize) {
    for (byte i = 0; i < bufferSize; i++) {
        Serial.print(buffer[i] < 0x10 ? " 0" : " ");
        Serial.print(buffer[i], DEC);
    }
}
}

```

Annexe 2 : Programme pour Arduino IDE.

```
#include <SPI.h>
#include <MFRC522.h>
#include <SoftwareSerial.h>
SoftwareSerial mySerial(7,8);
String textForSMS;
#define SS_PIN 10
#define RST_PIN 9
MFRC522 mfrc522(SS_PIN, RST_PIN);    // definie les PINs de Arduino qui controle le module
MFRC522.

void setup()
{
  mySerial.begin(9600); // Initialization de communications Serie avec le PC (USB port)
  Serial.begin(9600);// definie le baud rate de communication Serie (Arduino-PC)
  //Serial.println(" Hello Sir ! ");
  SPI.begin();          // Initialization de Bus SPI (protocole de communication de MFRC522)
  mfrc522.PCD_Init();   // Initialization de module MFRC522
}
////////////////////////////////////

void loop()
{
  if(Serial.available() == 0)
  {
    // Preparer le clé "key" et tout les "keys" sont de la form hexadecimale 'FFFFFFFFFFFFh'
    MFRC522::MIFARE_Key key;
    for (byte i = 0; i < 6; i++) {
      key.keyByte[i] = 0xFF;
    }
    // Lire une nouvelle Carte RFID:
    if (! mfrc522.PICC_IsNewCardPresent()) {
```

```

        return;
    }
    // Selection une de les Cartes deja Lire
    if ( ! mfr522.PICC_ReadCardSerial() ) {
        return;
    }
    // Affichage de ID carte en format Decimal au lieu de Hexadecimal
    Serial.print(" ID de la Carte :");
    for (byte i = 0; i < mfr522.uid.size; i++) {
        Serial.print(mfr522.uid.uidByte[i] < 0x10 ? " 0" : " ");
        Serial.print(mfr522.uid.uidByte[i], DEC);
    }
    Serial.println();
    // Dump PICC type
    byte piccType = mfr522.PICC_GetType(mfr522.uid.sak);
    if (    piccType != MFRC522::PICC_TYPE_MIFARE_MINI
        &&    piccType != MFRC522::PICC_TYPE_MIFARE_1K
        &&    piccType != MFRC522::PICC_TYPE_MIFARE_4K) {

        return;
    }
    // initialization des cartes RFID pour chaque eleve :
    // eleve A
    if( (mfr522.uid.uidByte[0] == 215) && (mfr522.uid.uidByte[1] == 32) &&
(mfr522.uid.uidByte[2] == 66) && (mfr522.uid.uidByte[3] == 115) ) // carte de élèveA
    {
        Serial.println(" eleveA");
        delay(5000);
    }

    // eleve B

```

```

    if( (mfr522.uid.uidByte[0] == 74) && (mfr522.uid.uidByte[1] == 5) &&
(mfr522.uid.uidByte[2] == 70) && (mfr522.uid.uidByte[3] == 207) ) // carte de élèveB
    {
        Serial.println(" eleveB");
        delay(5000);
    }
    else
        Serial.println("utilisateur non inscrit");
}
//////////Initialisation des SMS et les Data des parents//////////
if (Serial.available(>0)){
    char data = Serial.read();
    delay(1000);
    // Envoyer des sms au parents :
    // Interface commandes 'Visual Basic 6.0'
    if(data=='l'){
        textForSMS = "\n EleveA est absent"; // ce message envoyer au parents
        SendMessage1(textForSMS);
        Serial.println(textForSMS);
        Serial.println(" Message envoyer. "); // ce message afficher sur le Serial Moniteur
        data = 'z'; // changement de valeur de variable data pour evité la loop des sms
        delay(5000);
    }
    if(data=='m'){
        textForSMS = "\n EleveB est absent"; // ce message envoyer au parents
        SendMessage2(textForSMS);
        Serial.println(textForSMS);
        Serial.println(" Message envoyer. "); // ce message afficher sur le Serial Moniteur
        data = 'z'; // changement de valeur de variable data pour evité la loop des sms
        delay(5000);
    }
}

```

```

}
}
////////////////////////////////////
void SendMessage1(String message)
{
  mySerial.println("AT+CMGF=1"); // AT command pour envoyer le SMS
  delay(1000); // pause de 1000 milli seconds ou 1 second
  mySerial.println("AT+CMGS=\"+213xxxxxxxxx\"\\r"); // taper ici le Numero des parents de
l'Eleve 1
  delay(1000);
  mySerial.println(message); // message pour envoyer
  delay(100);
  mySerial.println((char)26); // terminer AT command avec ^Z, « ASCII code 26 = Control + z »
  delay(100);
  mySerial.println();
  delay(5000); // donner le temp au module d'envoyer le SMS
}
////////////////////////////////////
void SendMessage2(String message) {
  mySerial.println("AT+CMGF=1"); // AT command pour envoyer le SMS
  delay(1000); // pause de 1000 milli seconds ou 1 second
  mySerial.println("AT+CMGS=\"+213xxxxxxxxx\"\\r"); // taper ici le Numero des parents de
l'Eleve 2
  delay(1000);
  mySerial.println(message); // message pour envoyer
  delay(100);
  mySerial.println((char)26); // terminer AT command avec ^Z, ASCII code 26 (C-A-D: Control + z)
  delay(100);
  mySerial.println();
  delay(5000); // donner le temp au module d'envoyer le SMS
}

```

Annexe 3 : Programme de Visual Basic 6.0.

Option Explicit

Private Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)

Dim message As Integer

Dim search As String

Dim mins As Integer

Dim secs As Integer

Private Sub Form_Load()

lblDate.Caption = Format(Now(), "short date") ' La date afficher dans l'interface

mins = 0

Timer3.Enabled = True ' timer3 utiliser pour calculer les minutes et les seconds

'MSComm1.Settings = "9600,n,8,1"

'MSComm1.CommPort = 10

'MSComm1.PortOpen = True

With MSComm1

.CommPort = 4

.Settings = "9600,N,8,1"

.Handshaking = comRTS

.RTSEnable = True

.DTREnable = True

.RThreshold = 1

.SThreshold = 1

.InputMode = comInputModeText

.InputLen = 0

.PortOpen = True

End With

End Sub

Private Sub Timer1_Timer()

txtrec.Text = MSComm1.Input

```

search = txtrec.Text
If (InStr(search, "eleveA")) Then 'lecteur RFID lire ID et le nom d'élève A
Check1.Value = 1
End If
If (InStr(search, "eleveB")) Then 'lecteur RFID lire ID et le nom d'élève B
Check2.Value = 1
End If
End Sub

Private Sub Timer2_Timer()
Label5.Caption = Time ' Horloge (le temp afficher dans l'interface)
End Sub

Private Sub Timer3_Timer() 'calculer les mins et les secs et le tmp pour coché les absents
secs = secs + 1
Label7.Caption = secs ' afficher les seconds dans Label7 a l'interface
If secs > 59 Then
secs = 0
mins = mins + 1
Label3.Caption = mins ' afficher les minutes dans Label3 a l'interface
End If
If ((mins < 1) And (Check1.Value = 1)) Then
Check3.Value = 0
Else
If ((Check1.Value = 0) And (mins > 1)) Then 'si l'Eleve 1 venu apres 2 min , absent
Check3.Value = 1
End If
End If
If ((mins < 1) And (Check2.Value = 1)) Then
Check4.Value = 0
Else

```



```

If ((Check2.Value = 0) And (mins > 1)) Then 'si l'Eleve 2 venu apres 2 min , absent
Check4.Value = 1
End If
End If
txtMins.Text = mins
txtSecs.Text = secs
If ((txtMins.Text = "3") And (txtSecs.Text = "1")) Then
MsgBox " C'est le temp envoyer les SMS ", vbExclamation, Title:=" Alert ! "
End If
End Sub
Private Sub Command1_Click() 'ce Boutton est pour envoyer les SMS
If (Check3.Value = 1) Then
MSComm1.Output = "1" ' Envoyer a l'Arduino : DATA="1" pour SMS au parents d'élèveA
End If
If (Check4.Value = 1) Then
MSComm1.Output = "m" ' Envoyer a l'Arduino : DATA="m" pour SMS au parents d'élèveB
End If
End Sub
' PS : les numéros des parents sont stockés au niveau de croquis d'Arduino

```

Références Bibliographiques

- [1] K. Bellal, Z. Zaid et I. M, «conception et réalisation d'un système,» projet de fin d'étude, master, électronique industrielle, université de tizi-ouzou, 2016.
- [2] M. Boucebssi, A. Medjber et Lazri, «conception et réalisation d'un système,» projet de fin d'étude, master, électronique industriel, université de tizi-ouzou, 2017.
- [3] R. N. A. Randrianomanana et E. Herinantenaina, «pilotage a distance par,» Mémoire en vue de l'obtention du diplôme de Master, électronique automatique, université de antananarivo, 2016.
- [4] <https://www.comelitgroup.com/fr-fr/systemes/domotique/les-fonctions/>. [accès le 07 05 2021].
- [5] <https://knx-automation.com/domotique/les-fonctions-domotiques-de-la-maison-intelligente/>. [accès le 07 05 2021].
- [6] I. j et h. c, écrivains, la domotique. les maisons connectées, comment la technologie peut-elle faciliter notre quotidien?. communaute francaise, 2017.
- [7] <https://www.positron-libre.com/electronique/arduino/arduino.php>. [accès le 28 04 2021].
- [8] S. Ait Allouche, I. Mazari et O. F, «conception et réalisation d'un réseau de stations,» projet de fin d'étude, master électronique, instrumentation, université de tizi-ouzou, 2018.
- [9] <https://www.gotronic.fr/art-carte-arduino-esplora-20518.htm>. [accès le 13 06 2021].
- [10] https://www.gotronic.fr/art-carte-arduino-micro-19537.htm#complte_desc. [accès le 10 06 2021].
- [11] <https://www.gotronic.fr/cat-modules-lilypad-1428.htm>. [accès le 16 06 2021].
- [12] J. Nussey, D. Duplan et S. Bontemps, arduino pour les nuls, paris: éditions first, 2017.
- [13] <https://bentek.fr/2-arduino-uno/>. [accès le 29 05 2021].
- [14] <http://arduino.blaisepascal.fr/presentation/logiciel-ide-arduino/>. [accès le 12 05 2021].
- [15] <https://www.companeo.com/portiques-de-securite/guide/technologie-rfid>. [accès le 01 05 2021].
- [16] <https://www.companeo.com/portiques-de-securite/guide/puces-rfid-tag-badge-etiquette>. [accès le 01 05 2021].
- [17] <https://www.companeo.com/portiques-de-securite/guide/lecteur-rfid>. [accès le 01 05 2021].
- [18] <https://www.electronique-mixte.fr/microcontrolleurs/rfid-contrôle-d'accès-par-badge-avec-arduino>. [accès le 02 06 2021].

- [19] H. Bendjafer, A. Midoun et A. Nemmiche, «système de contrôle par gsm,» projet fin d'étude, master, instrumentation électronique, université de tlemcen, 2017.
- [20] E. E. O. Astalaseven, arduino pour bien commencer en électronique et en programmation, paris: le site du zéro, 2012.
- [21] <https://www.aranacorp.com/fr/utilisation-dun-module-rfid-avec-arduino>. [accès le 18 06 2021].