



# جامعة أبو بكر بلقايد- تلمسان

**Université Abou Bakr Belkaïd de Tlemcen**

**Faculté de Technologie**

**Département de Génie Biomédical**

**MEMOIRE DE PROJET DE FIN D'ETUDES**

**Pour l'obtention du Diplôme de**

**MASTER en GENIE BIOMEDICAL**

**Spécialité : Instrumentation Biomédical**

**Présenté par : SAIDI Widad et NAIM Fatima Zahra**

---

## Conception et réalisation d'un lecteur en Braille

---

**Présenté le 30 Septembre 2020 devant les Jurys**

<b>Mr.</b>	<b>S.M. Debbal</b>	<b>Université de Tlemcen</b>	<b>Président</b>
<b>Mr.</b>	<b>Sid Ahmed Taouli</b>	<b>Université de Tlemcen</b>	<b>Examineur</b>
<b>Mr.</b>	<b>M.E.A. Brixi Nigassa</b>	<b>Université de Tlemcen</b>	<b>ENCADRANT</b>
<b>Mr.</b>	<b>Ahmed Slami</b>	<b>Université de Tlemcen</b>	<b>CO-ENCADRANT</b>

**Année universitaire 2019-2020**

*À nos chers parents, À nos sœurs et frères*  
*À toutes personnes non-voyante*

## *Remerciements*



*Tout d'abord et avant tout, nous remercions **ALLAH** de tout puissant qui nous a donné la force, la volonté, la patience et le courage pour finaliser ce modeste travail*

*Nous remercions, notre encadrant **Dr. BRIXI NIGASSA MOHAMMED EL AMINE**.*

*Pour son encadrement de grande qualité scientifique, ses précieux conseils, et son encouragement ; qui ont contribué efficacement à l'avancement de ce travail.*

*Nos sincères remerciements vont à **Mr. SLAMI AHMED** qui, en tant qu'un Co-Encadrant de mémoire, il est toujours montré à l'écoute et très disponible tout au long de la réalisation de ce mémoire, ainsi pour l'inspiration, l'aide et le temps qu'il est bien voulu nous consacrer*

*Nous adressons de même nos remerciements à **Mr. Debbal** et **Mr. Taouli** , pour l'intérêt qu'ils ont accordé à ce travail en acceptant de présider et d'examiner notre modeste travail*

*Nous tenons à exprimer nos gratitude et notre profonde reconnaissance à **nos chers parents**, à toutes nos familles nos amis et à tous ceux qui nous ont aidé de près en de loin tout au long de notre travail.*

***Merci à tous et à toutes***





## Résumé

L'objectif de ce travail est de proposer un dispositif pour la lecture en Braille destiné aux personnes non-voyantes, simple d'utilisation et peu coûteux. Le Braille se présente sous la forme d'une matrice de six points dont la combinaison permet de donner un chiffre, une lettre, un caractère spécial. Ce dernier est régi par des standards internationaux présents dans les pays développés mais pas en Algérie. Ces standards déterminent les dimensions et seuils de sensibilité chez des personnes non-voyantes. Ainsi, une partie de ce travail a consisté à élaborer une petite base de données permettant de déterminer ces seuils et dimensions. Dans la deuxième partie, une application Android (BrailleApp) a été créée. Cette dernière permet de traduire un SMS reçu sur un smartphone en code de Braille pour l'afficher sur un lecteur en Braille. Ce dernier est basé sur une carte Arduino sur laquelle est connecté un module Bluetooth pour assurer la communication avec le smartphone. Les six points de la matrice en Braille sont quant à eux représentés par six servomoteurs se déplaçant avec une hauteur de 400µm afin de permettre une bonne perception de cette matrice de Braille par l'individu non-voyant. Les résultats obtenus sont très satisfaisants et prometteurs permettant d'envisager de pousser encore le concept beaucoup plus loin, en ne se limitant pas uniquement à la lecture en Braille d'un SMS.

**Mots clés :** Personne non-voyante, Braille, standards, code Braille, lecteur en Braille, SMS, Smartphone, Application Android.

## Abstract

This work aims to provide a device for Braille reading intended for blind people, easy to use and inexpensive. Braille is in the form of a matrix of six dots, the combination of which gives a number, a letter, a special character. The latter is governed by international standards present in developed countries but not in Algeria. These standards determine the dimensions and thresholds of sensitivity for blind people. Thus, the first part of this work consisted of building a small database to determine these thresholds and dimensions. In the second part, an Android application (BrailleApp) was created. This latter allows translating an SMS received on a smartphone into Braille code which is displayed on a Braille reader. The latter is based on an Arduino board to which a Bluetooth module is connected to ensure communication with the smartphone. The six points of the Braille matrix are represented by six servo motors moving with a height of 400 µm to allow a good perception of this Braille matrix by the blind people. The results obtained are very satisfying and promising, making it possible to consider pushing the concept even further, by not limiting ourselves only to an SMS reading in Braille.

**Keywords :** Blind people, Braille, standards, Braille code, Braille reader, SMS, Smartphone, Android application.

## ملخص

الهدف من هذا العمل هو توفير جهاز لقراءة برايل مخصص للمكفوفين، سهل الاستخدام وغير مكلف. تأخذ طريقة برايل شكل مصفوفة متكونة من ست نقاط، يعطي الجمع بينها رقمًا أو حرفًا أو رمزًا خاصًا. هذا الأخير تحكمه المعايير الدولية الموجودة في البلدان المتقدمة ولكن للأسف ليس في الجزائر. تحدد هذه المعايير الأبعاد وحدود الحساسية لدى المكفوفين. وبالتالي، فإن الجزء الأول من هذا العمل يعتمد على بناء قاعدة بيانات صغيرة لتحديد هذه العتبات والأبعاد. في الجزء الثاني، تم إنشاء تطبيق اندرويد (BrailleApp). يسمح هذا الأخير ترجمة التي تتصل بها Arduino الرسائل القصيرة المستلمة على الهاتف الذكي إلى رمز برايل لعرضه على آلة البرايل. يعتمد هذا الأخير على لوحة لضمان الاتصال بالهاتف الذكي. يتم تمثيل النقاط الستة لمصفوفة برايل بستة محركات مؤازرة تتحرك بارتفاع 400 Bluetooth وحدة ميكرومتر من أجل السماح للفرد الكفيف بالإدراك الجيد لمصفوفة برايل. النتائج التي تم الحصول عليها مرضية للغاية وواعدة، مما يجعل من الممكن التفكير في دفع المفهوم إلى أبعد من ذلك، من خلال عدم الإقتصار على قراءة الرسائل القصيرة بطريقة برايل فقط.

**الكلمات المفتاحية:** برايل، المكفوفين، معايير، آلة البرايل، الرسائل القصيرة، هاتف ذكي، رمز برايل، دائرة الكترونية، قاعدة بيانات صغيرة، تطبيق أندرويد

# Table des matières

Introduction générale.....	12
Chapitre I : « Le code de Braille ».....	13
I.1. Introduction :.....	14
I.2. Historique :.....	14
I.3. C'est quoi le braille ?.....	15
I.4. L'alphabet et les chiffres en braille :.....	16
I.5. Les standards du braille dans le monde :.....	16
I.6. Lecteurs en braille :.....	17
I.7. Conclusion :.....	20
Chapitre II : La conception matériel, logiciel et simulation .....	21
II.1. Introduction :.....	22
II.2. Concept du lecteur en Braille :.....	22
II.3. Partie hardware :.....	22
II.3.1. La carte Arduino :.....	22
II.3.1.1. IDE (Integrated Development Environment):.....	23
II.3.2. Module Bluetooth HC-05 :.....	24
II.3.3. Les Servomoteurs :.....	25
II.3.3.1. Les caractéristiques du servomoteur SG90 :.....	25
II.3.3.2 Connexion du servomoteur avec la carte Arduino UNO :.....	26
II.4. Partie application Android :.....	26
II.4.1 App Inventor :.....	27
II.4.1.1 Environnement App Inventor :.....	27
II.5. Simulation sous « PROTEUS ISIS » :.....	29
II.5.1. Présentation du logiciel 'PROTEUS' :.....	30
II.5.2. Simulations sous PROTEUS ISIS :.....	30
II.5.2.1. Simulation avec LED :.....	30
II.5.2.2. Simulation avec servomoteurs :.....	30
II.6. Conclusion :.....	31
Chapitre III : Réalisation du lecteur en Braille .....	32
III.1. Introduction :.....	33
III.2. Etude du sens de toucher en Algérie :.....	33
III.2.1. Modélisation sous Autodesk Inventor :.....	34
III.2.2. Simplify 3D :.....	34
III.2.3. Imprimante 3D :.....	35

<b>III.2.4. Le test de toucher :</b> .....	35
<b>III.3. Réalisation du lecteur en Braille :</b> .....	36
<b>III.3.1 Application Android :</b> .....	36
<b>III.3.2. Lecteur en Braille en utilisant des LED :</b> .....	40
<b>III.3.3. Lecteur en Braille en utilisant des servomoteurs :</b> .....	43
<b>Conclusion générale et perspectives :</b> .....	46
<b>Références :</b> .....	47
<b>Annexes :</b> .....	1



# Liste des figures

## Chapitre I :

<b>Fig.I. 1</b> Système d'écriture de Valentine Haüy [4] .....	14
<b>Fig.I. 2</b> Sonographie de Charles Barbier [5].....	15
<b>Fig.I. 3</b> Cellule Braille .....	15
<b>Fig.I. 4</b> Le Braille sur différents aspects du quotidien [6], [7] .....	16
<b>Fig.I. 5</b> L'alphabet et les chiffres en code de Braille.....	16
<b>Fig.I. 6</b> Dimensions régissant les standards du code de Braille .....	17
<b>Fig.I. 7</b> Top Braille [9] .....	18
<b>Fig.I. 8</b> Plage Braille [10] .....	18
<b>Fig.I. 9</b> Dot Smart Watch [11].....	18
<b>Fig.I. 10</b> Cellule braille piézoélectrique [12] .....	19
<b>Fig.I. 11</b> Membranes piézoélectriques [13] .....	19
<b>Fig.I. 12</b> BrailleBox [14] .....	19
<b>Fig.I. 13</b> Matrices de Braille pneumatiques [15] .....	20

## Chapitre II :

<b>Fig.II. 1</b> Les composants du lecteur en braille proposé.....	22
<b>Fig.II. 2</b> Eléments constituant de la carte Arduino.....	23
<b>Fig.II. 3</b> L'interface de programmation Arduino (IDE). .....	23
<b>Fig.II. 4</b> Le module Bluetooth HC-05 [17] .....	24
<b>Fig.II. 5</b> Connexion entre le module HC-05 et la carte Arduino.....	24
<b>Fig.II. 6</b> Types des servomoteurs [18] .....	25
<b>Fig.II. 7</b> Un servomoteur SG90 .....	25
<b>Fig.II. 8</b> Connexion d'un servomoteur SG90 avec la carte Arduino UNO.....	26
<b>Fig.II. 9</b> Fenêtre de création de l'interface de l'application Android sous App Inventor.....	27
<b>Fig.II. 10</b> La fenêtre de programmation sous App Inventor .....	27
<b>Fig.II. 11</b> Fenêtre de choix pour le test de l'application Android .....	28
<b>Fig.II. 12</b> MIT AI Companion.....	28
<b>Fig.II. 13</b> L'émulateur Android aiStarter .....	29
<b>Fig.II. 14</b> Méthodes d'installation de l'application Android .....	29
<b>Fig.II. 15</b> Simulation de la matrice de Braille avec six LED (lettre H en Braille) .....	30
<b>Fig.II. 16</b> Simulation de la matrice de Braille avec six servomoteurs (lettre H en Braille) .....	31

## Chapitre III :

<b>Fig.III. 1</b> Schéma bloc de la réalisation pratique.....	33
<b>Fig.III. 2</b> Structure modélisée sous Autodesk Inventor.....	34
<b>Fig.III. 3</b> Scène d'impression du logiciel Simplify3D .....	34
<b>Fig.III. 4</b> Imprimante 3D Printrbot Metal Plus.....	35
<b>Fig.III. 5</b> Structures imprimées pour le test du toucher .....	35
<b>Fig.III. 6</b> Schéma de fonctionnement du lecteur en braille .....	36
<b>Fig.III. 7</b> L'interface de l'application Android créée .....	37

<b>Fig.III. 8</b> Programmation de l'application Android .....	37
<b>Fig.III. 9</b> Etapes pour tester et installer l'application "BrailleApp" .....	38
<b>Fig.III. 10</b> L'installation de l'application en utilisant le QR code. ....	39
<b>Fig.III. 11</b> Connexion Bluetooth entre smartphone et module HC-05 .....	40
<b>Fig.III. 12</b> Schéma du lecteur en Braille utilisant des LED .....	41
<b>Fig.III. 13</b> Affichage de la lettre "b" en Braille.....	41
<b>Fig.III. 14</b> Schéma du lecteur en Braille utilisant des servomoteurs .....	43

## Liste des tableaux

<b>Tableau 1.</b> Standards de Braille dans quelques pays [8] .....	17
<b>Tableau 2.</b> Les caractéristiques du servomoteur SG90[18] .....	26
<b>Tableau 3.</b> L’affichage des deux mots ‘merci’ et ‘labo gbm’ en code de Braille respectivement sur notre lecteur en Braille.....	42

## **Introduction générale**

## Introduction générale

Les organes sensoriels jouent un rôle primordial dans notre vie quotidienne. Grâce à ces derniers, nous pouvons acquérir différents types d'informations sur le monde qui nous entoure. On distingue ainsi cinq sens : le goût (gustation), l'odorat (olfaction), l'ouïe (audition), la vue (vision) et le toucher.

Pour les personnes en pleine possession de leurs cinq sens, l'accès à une information quelconque n'est pas un problème. En revanche, pour des personnes présentant un handicap dans l'un des cinq sens, l'accès à une information même simple devient très vite un obstacle au quotidien. Dans notre projet de fin d'étude, nous nous intéressons aux personnes présentant une déficience ou cécité visuelle (les non-voyants). Pour ces personnes, plusieurs moyens ont été mis en œuvre pour réduire ce handicap. La plupart de ces outils sont basés sur le sens du toucher et ont pour support le code de Braille. Ce dernier est un code permettant de représenter une lettre, un chiffre ou un caractère spécial dans une matrice de six points appelée matrice de Braille. En changeant la combinaison des points de la matrice, nous changeons le caractère à afficher. Le Braille est régi par des standards qui sont différents d'une région (ou pays) à une autre et que l'on retrouve dans les pays développés. Malheureusement en Algérie, aucun standard n'existe et de ce fait, une des parties de ce travail consistera à établir une petite base de données permettant de définir un seuil de sensibilité de toucher perceptible par des Algériens.

A partir de là, nous entamerons la deuxième partie où il sera question de réaliser un lecteur en Braille permettant de traduire un texte en code Braille, simple à utiliser, peu encombrant et surtout à faible coût en exploitant des techniques simples et à l'air du temps. Pour ce faire, une application de reconnaissance de texte sera introduite afin d'envoyer ce texte à un circuit électronique permettant l'affichage des caractères transcrits en Braille. L'une des solutions envisagées est d'utiliser six actionneurs représentant les six points de la matrice de Braille. Ayant déjà une information sur la perception du toucher chez un Algérien (seuil défini dans la première partie), nous pourrions faire surélever ces actionneurs avec une hauteur parfaitement perceptible et contrôlable.

Nous organisons notre travail de recherche en trois chapitres :

Dans le premier chapitre, nous donnons un aperçu général sur l'historique du braille et sa définition, puis nous aborderons l'alphabet du braille et ses standards dans le monde. Enfin, nous présenterons des lecteurs en Braille qui existent déjà et quelques travaux de recherche sur le Braille.

Le deuxième chapitre présente les éléments électroniques et les logiciels de programmation dont nous avons besoin dans notre travail, tels que : la carte Arduino avec son logiciel de programmation IDE, le module Bluetooth, les servomoteurs et l'application Android. Les caractéristiques, les avantages et le principe de fonctionnement de chaque élément seront illustrés. Enfin, Le montage de notre circuit sera simulé en utilisant le logiciel PROTEUS.

Dans le troisième et le dernier chapitre, nous allons présenter l'étude et les résultats obtenus du test de toucher dans notre région en créant une base de données. Ensuite, nous passerons à détailler la réalisation pratique du lecteur en commençant par la création d'une application Android qui envoie un SMS, reçu sur un smartphone, vers le lecteur afin d'afficher son transcription en code de braille. Enfin, nous allons présenter le montage du circuit et le test sur des LEDs afin de passer à la réalisation finale de notre lecteur basé sur les servomoteurs.

Nous terminerons ce mémoire par une conclusion générale qui résume l'ensemble de nos travaux réalisés ainsi que les perspectives envisageables dans la continuité de ce travail.

## **Chapitre I : « Le code de Braille »**

## I.1. Introduction :

Selon l'Organisation Mondiale de la Santé (OMS), il y aurait 135 millions de personnes souffrant de déficience visuelle, 40 à 45 millions d'entre elles sont en cécité visuelle en 2020 [1]. En Algérie, plus de 173000 personnes souffrent de cécité visuelle selon le dernier rapport de l'Organisation Nationale de la Santé (ONS) [2].

Pour faciliter le quotidien des personnes non-voyantes et leur permettre l'accès à une information donnée, plusieurs techniques ont été mise en œuvre parmi lesquelles on trouve : Système d'écriture de Valentine Haüy, Sonographe de Charles Barbier et le code de Braille. Néanmoins, c'est le code de Braille reste le plus utilisé de nos jours. Ce code utilise une matrice appelée « matrice braille » composée de points en reliefs dont la hauteur vient changer la perception du toucher chez les personnes non-voyantes.

Nous commencerons ce chapitre par un petit historique sur l'invention du Braille. Nous verrons par la suite les différents standards internationaux régissant le Braille, avant de citer quelques lecteurs en braille existant dans le marché.

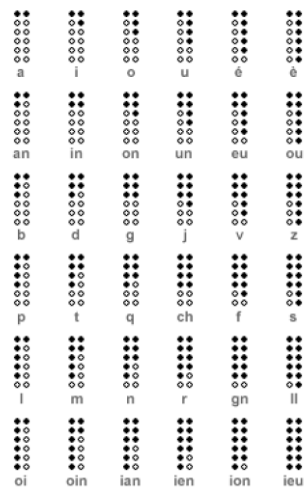
## I.2. Historique :

Le code de Braille a été inventé par Louis Braille en 1829, devenu aveugle à l'âge de 3 ans à cause d'un accident dans l'atelier de son père. Louis Braille a fait ses études en 1819 à l'Institut Nationale des Jeunes Aveugles (INJA) qui a été fondé par Valentin Haüy en 1786 [3]. En fondant cet institut, Valentin Haüy avait pour objectif de permettre aux aveugles d'accéder à la lecture. Pour cela, il fit réaliser un code sous forme des lettres d'alphabet en grand format (Fig.I.1) et s'en est servi pour gaufrer le papier, Louis Braille apprenait à lire grâce à ce code [4].



**Fig.I. 1** Système d'écriture de Valentine Haüy [4]

En 1821, Louis Braille fit la connaissance de Charles Barbier de la Serre qui avait inventé un procédé d'écriture des sons en relief appelé « Sonographie », représentant 12 caractères en reliefs (Fig.I.2) permettent d'écrire différents sons [5].

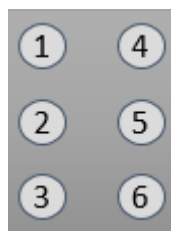


**Fig.I. 2** Sonographie de Charles Barbier [5]

Louis Braille propose alors à Barbier quelques améliorations de son système de Sonographie, des améliorations que Barbier ne prendra jamais en considération. Louis Braille se lance alors seul dans l'élaboration d'un nouveau code. Après quelques années de recherches, il mit au point l'écriture orthographique en code telle qu'on la connaît aujourd'hui et qui porte son nom : code de Braille.

### I.3. C'est quoi le braille ?

Le braille est une petite matrice, appelée cellule, de 6 points disposés sur trois lignes et deux colonnes (Fig.I.3). La combinaison de ces points permet non seulement de représenter l'alphabet, mais également des chiffres, des signes de ponctuation et même des signes mathématiques.



**Fig.I. 3** Cellule Braille

Le Braille est présent sous différentes formes dans notre quotidien, on le trouve partout, dans les ascenseurs, les boîtes pharmaceutiques, des ouvrages imprimés en Braille, plaques signalétiques, etc.... (Fig.I.4). Grâce à ce code de Braille, la personne non-voyante peut lire un livre, choisir l'étage d'un ascenseur et identifier ses médicaments.

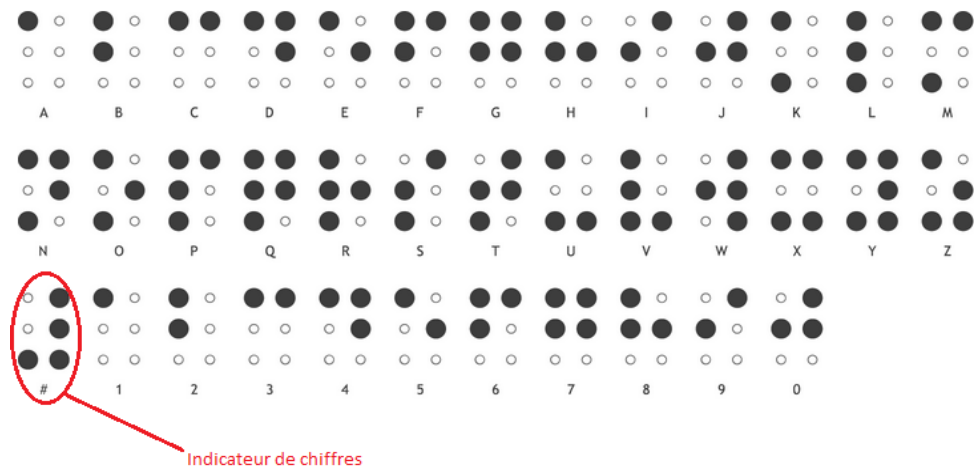


**Fig.I. 4** Le Braille sur différents aspects du quotidien [6], [7]

Nous allons définir dans ce qui suit l’alphabet en Braille qui nous permettra justement d’identifier quel caractère nous sommes en train de lire.

### I.4. L’alphabet et les chiffres en braille :

La figure I.5 donne la représentation des caractères de l’alphabet et des chiffres dans le code de Braille.



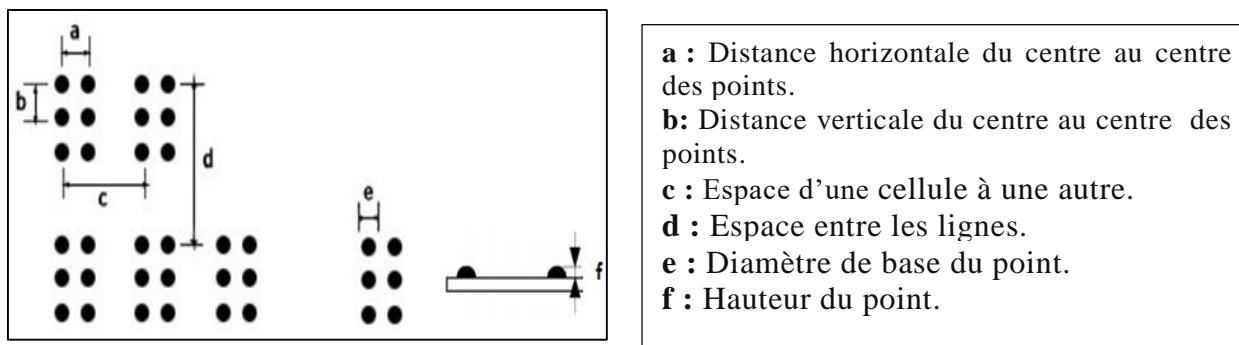
**Fig.I. 5** L’alphabet et les chiffres en code de Braille

Les points en noir sur la figure.I.5 représentent des points en relief, c’est-à-dire ayant une certaine hauteur, alors que les points en blanc ont une hauteur nulle.

### I.5. Les standards du braille dans le monde :

Le braille est régi par des standards permettant de définir la hauteur, les dimensions ainsi que l’espace relatifs aux points de la matrice afin qu’elle soit perceptible par la personne non-voyante (Fig.I.6).





**Fig.I. 6** Dimensions régissant les standards du code de Braille

Ainsi, en fonction de la région dans le monde où l'on se trouve, ces dimensions changent. Le tableau 1 représente les standards de braille dans quelques pays :

Pays	a(mm)	b(mm)	c(mm)	d(mm)	e(mm)	f(mm)
France	2.5-2.6	2.5-2.6	/	>10	1.2	0.8-1
USA	2.34	2.34	6.22	10.16	1.45	0.48
Allemand	2.5	2.5	6	10	1.3-1.6	0.5
Italie	2.2-2.5	2.2-2.5	/	/	1	0.5
Corée	2	2	5	6	1.5	0.6
Suède	2.5	2.5	6	10	1	0.25
Algérie	NC	NC	NC	NC	NC	NC

**Tableau 1.** Standards de Braille dans quelques pays [8]

Comme nous pouvons le constater sur le tableau 1, il n'existe pas de standard régissant le braille en Algérie. Par ailleurs, une partie de notre travail présentée dans le chapitre III sera de définir quelles sont les dimensions à retenir pour qu'une personne non-voyante vivant en Algérie puisse percevoir des caractères en Braille.

## I.6. Lecteurs en braille :

Il existe plusieurs dispositifs commerciaux permettant la lecture en Braille. La plupart exploitent le principe de l'actionnement électromécaniques pour afficher des caractères en Braille. Nous avons choisi de citer trois exemples de dispositifs commerciaux.

- **Le Top Braille :**

C'est un appareil doté d'une caméra numérique qui prend des photos de la surface sur laquelle il est posé. Ces images sont ensuite numérisées et transmises au processeur pour être traitées et convertir le texte en code de braille [9](Fig.I.7).



**Fig.I. 7** Top Braille [9]

- **Plage Braille :**

C'est un dispositif connecté par Bluetooth ou par un port USB permettant la conversion d'un texte présent à l'écran en code de braille sur des cellules de braille disposé sur la surface d'un clavier. Ce dernier permet aussi la saisie d'un texte et dispose de touches pour naviguer dans l'écran [10]. (Fig.I.8)



**Fig.I. 8** Plage Braille [10]

- **Dot Smart Watch :**

Cette montre permet à une personne non-voyante de recevoir des informations via son téléphone en temps réel, telles que les notifications, les messages, etc... [11]( Fig.I.9).



**Fig.I. 9** Dot Smart Watch [11]

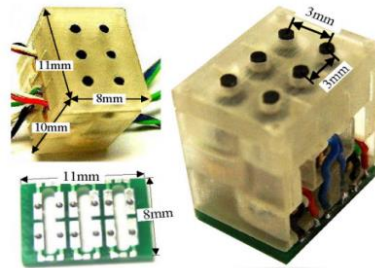
Même si ces dispositifs sont perfectionnés et présentent différentes options, ils restent néanmoins pour la plus part lourds, encombrants et surtout hors de prix sachant que 90% des personnes non-voyantes à travers le monde vivent dans des pays à faible revenu [1].

Plusieurs travaux de recherches ont été menées justement pour palier entre autres à ce problème de coût, mais aussi afin de concevoir des systèmes de plus en plus compacts et ergonomique en faisant

appel à différentes technologies d'actionnement. Quelques travaux de recherches sont cités dans la partie suivante.

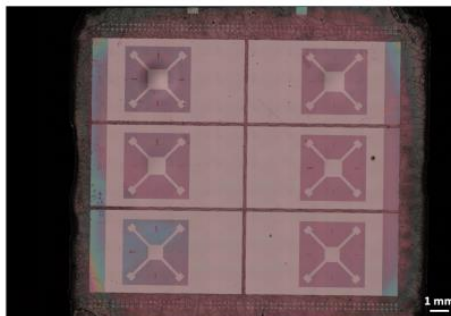
- **Matrices de braille piézoélectriques :**

En 2017, Un groupe d'ingénieurs ont développés un système de contrôle de la cellule de Braille piézoélectrique à l'aide d'un microcontrôleur PIC [12] (Fig.I.10) .



**Fig.I. 10** Cellule braille piézoélectrique [12]

En 2018, Brix Nigassa M.E.A et al. ont développés un procédé de réalisation des membranes piézoélectriques sous forme d'une cellule de braille. Ces dernières sont des micro-actionneur (MEMS) à forte amplitude de déflexion [13](Fig.I.11).



**Fig.I. 11** Membranes piézoélectriques [13]

- **Matrices de braille électromagnétiques :**

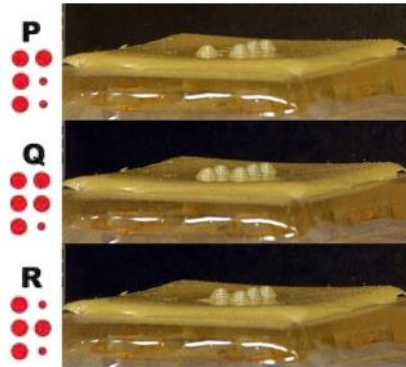
Joe Birch a construit un appareil simple qui convertit les nouvelles en ligne en braille, inspiré par la prédisposition de sa famille à la perte de la vue [14] (Fig.I.12).



**Fig.I. 12** BrailleBox [14]

- **Matrices de Braille pneumatiques :**

Cette dernière est basée sur une série de valves et d'un réseau de microbulles agissant comme des points dans la cellule de Braille (Fig.I.13)[15].



**Fig.I. 13** Matrices de Braille pneumatiques [15]

Malgré tous les progrès technologiques, on se retrouve souvent soit avec des dispositifs commerciaux hors de prix et peu ergonomiques, soit avec de nouvelles technologies d'actionnement pas encore commercialisées. Ainsi, le but de ce travail est justement d'apporter une solution simple et peu coûteuse pour la lecture en Braille, c'est ce que nous allons justement détailler dans la deuxième et troisième partie de ce travail.

## **I.7. Conclusion :**

Dans ce chapitre, nous avons commencé par un petit historique sur Louis Braille et le code qu'il a inventé. Nous avons ensuite défini le code de Braille et sous quelle forme se présentait ce dernier. Quelques standards ont été ensuite cités avant de voir quelques exemples de dispositifs commerciaux et quelques travaux de recherches sur les actionneurs pour le code de Braille. A partir de là, nous avons constaté que les dispositifs commerciaux, même s'ils sont évolués et présentent différentes options restent hors de prix et peu ergonomiques. Différents travaux de recherches sont ou ont été menés justement afin d'apporter une réponse technologique pour réduire entre autres les coûts. Dans cette optique, nous avons imaginé un système simple et peu coûteux permettant la lecture en Braille, c'est justement ce que nous allons détailler dans les parties qui suivent à commencer par la présentation des différents outils qui nous ont permis de mener à bien ce travail.

## **Chapitre II : La conception matériel, logiciel et simulation**

## II.1. Introduction :

Après avoir défini le Braille et établi le contexte et la problématique de notre travail, nous allons voir les différents outils qui nous permettront de réaliser notre lecteur en Braille. Après la présentation du concept de notre lecteur en braille, nous détaillerons les outils électroniques et logiciels nécessaires à la réalisation de ce dernier. Nous montrerons ensuite le principe de création d'une application Android et nous terminerons par la simulation sous Proteus.

## II.2. Concept du lecteur en Braille :

Le concept proposé se compose de quatre composants essentiels (Fig.II.1) : carte Arduino, interface Bluetooth, servomoteurs et l'application Android. L'objectif ici vise à créer une application Android qui reçoit un SMS sur un smartphone et de l'envoyer via le module Bluetooth vers la carte Arduino. Cette dernière fera ensuite la transcription des caractères reçus par sms en code Braille. Enfin, ce code sera affiché sur des servomoteurs qui sont commandés par la carte Arduino.

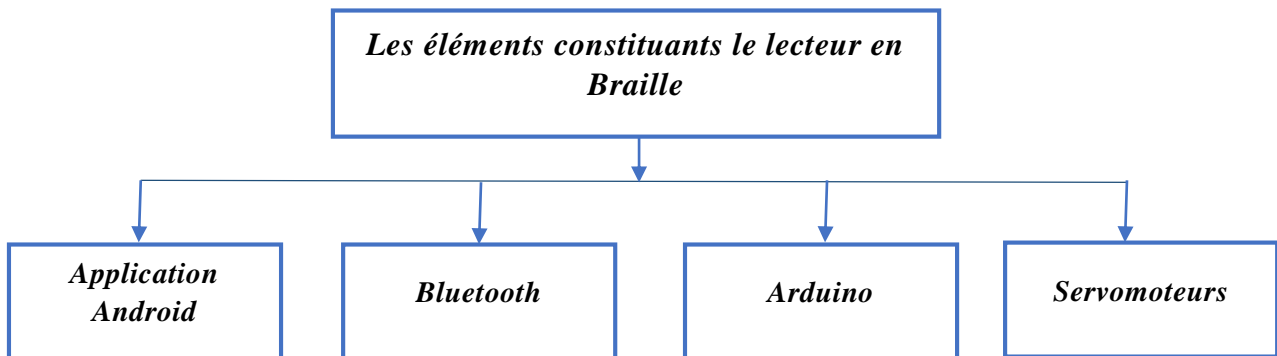


Fig.II. 1 Les composants du lecteur en braille proposé

Dans ce qui suit, nous allons détailler les différents composants de notre lecteur en Braille. Pour ce faire, nous avons divisé ces composants en deux parties : Partie Hardware et Partie application pour smartphone.

## II.3. Partie hardware :

Dans cette partie, il s'agira de définir les différents composants électroniques nécessaire au bon fonctionnement de notre lecteur en Braille.

### II.3.1. La carte Arduino :

L'Arduino est une plateforme de prototypage électronique open-source. Elle intègre deux systèmes : software (logiciel) et hardware (matériel). La carte Arduino possède de nombreux composants dont le plus important est le microcontrôleur [16]. Dans ce travail, nous avons choisi d'utiliser la carte Arduino UNO.

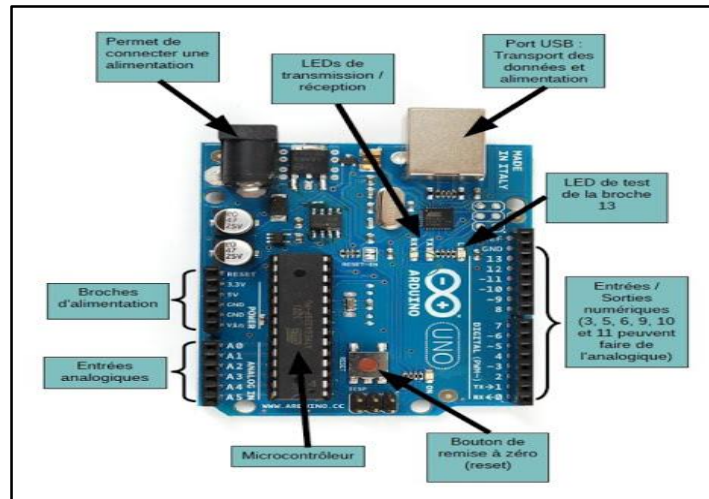


Fig. II. 2 Eléments constituant de la carte Arduino

### II.3.1.1. IDE (Integrated Development Environment):

L'IDE est un logiciel JAVA multiplateforme permettant d'éditer un code et de le compiler. Cet IDE comporte les éléments (icônes) suivants (Fig.II.3) :

- « Compiler » : pour la vérification du programme après édition.
- « Téléverser » : pour envoyer le programme vers la carte.
- Un espace pour éditer un programme.
- Une zone de message de compilation pour vérifier si le code ne contient pas d'erreurs.
- « Moniteur série » : permet la communication entre la carte Arduino et l'IDE.



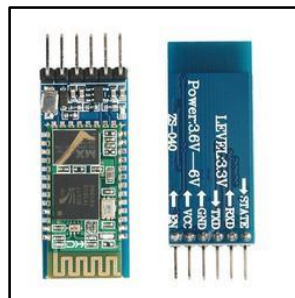
Fig. II. 3 L'interface de programmation Arduino (IDE).

En plus d'une carte Arduino, un module Bluetooth sera nécessaire afin d'assurer la communication entre le smartphone et la carte Arduino.

### II.3.2. Module Bluetooth HC-05 :

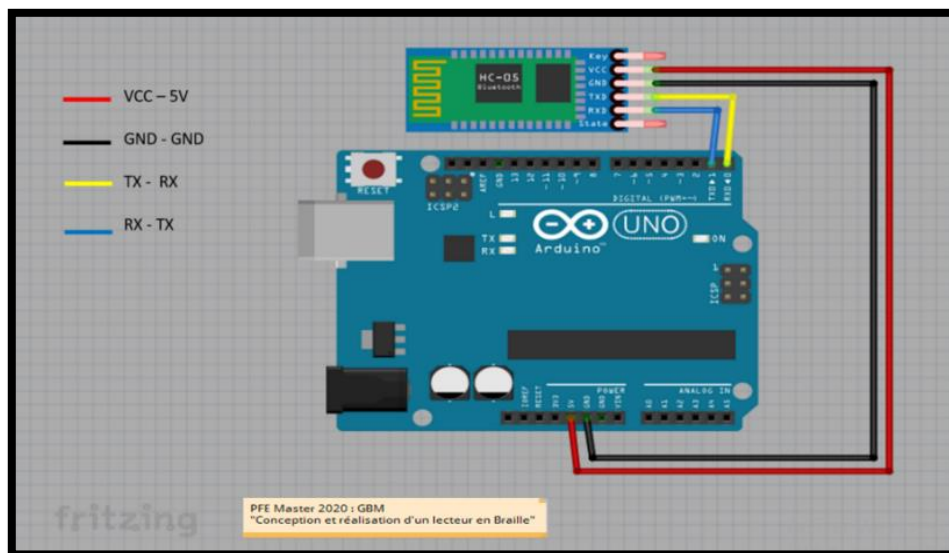
Le Bluetooth est une norme de communications permettant l'échange de données à courte distance en utilisant des ondes radio fréquences sur une bande de fréquence de 2.4GHz. Son utilisation permet de simplifier les connexions entre les appareils électroniques en supprimant les liaisons filaires. Il existe deux composants Bluetooth pour la carte Arduino : HC-05 et HC-06 [17].

Dans notre travail nous avons choisi le Bluetooth HC-05 pour assurer une communication bidirectionnelle « émission-réception » entre le smartphone et la carte Arduino (Fig.II.4).



**Fig.II. 4** Le module Bluetooth HC-05 [17]

La connexion du module Bluetooth HC-05 avec la carte Arduino est représentée sur la figure II.5.



**Fig.II. 5** Connexion entre le module HC-05 et la carte Arduino.

Nous pouvons aussi trouver deux autres broches dans ce module HC-05 :

- La broche State : est connectée avec une LED pour vérifier si le Bluetooth fonctionne correctement.
- La broche EN : doit être alimentée pour entrer dans le mode de configuration.



Pour finir, des servomoteurs représentant les points de la matrice de Braille seront banchés à la carte Arduino.

### II.3.3. Les Servomoteurs :

Le servomoteur appartient au groupe des moteurs électriques qui permettent de contrôler avec précision l'angle, l'accélération et la vitesse. Les servomoteurs se différencient en termes de taille et de puissance mais présentent le même principe de fonctionnement [18] (Fig.II.6).



Fig.II. 6 Types des servomoteurs [18]

Le servomoteur qui nous intéresse dans ce travail est le SG-90, ses caractéristiques sont présentées dans la partie suivante.

#### II.3.3.1. Les caractéristiques du servomoteur SG90 :

Ces moteurs tournent sur un axe suivant un angle allant de 0 à 180° avec une faible puissance. Nous avons choisi ce type de moteur de par sa petite taille mais aussi parce qu'ils peuvent générer précisément l'angle de rotation que l'on souhaite obtenir (Fig.II.7).

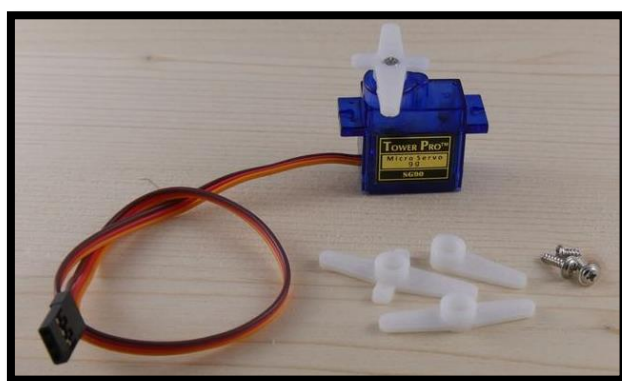


Fig.II. 7 Un servomoteur SG90

Les caractéristiques techniques du servomoteur SG90 sont données dans le tableau 2 qui suit :

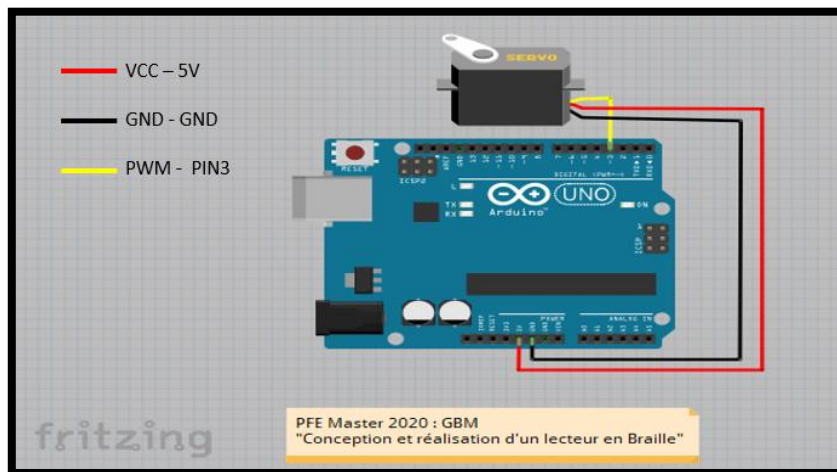
<b>La tension d'alimentation</b>	4.8V à 6 V
<b>Poids</b>	9 grammes
<b>Dimensions</b>	22*11.5*27 mm
<b>Vitesse</b>	0.12s / 60° sous 4.8V
<b>Couple</b>	1.2kg/cm sous 4.8V

**Tableau 2.** Les caractéristiques du servomoteur SG90[18]

### II.3.3.2 Connexion du servomoteur avec la carte Arduino UNO :

Le câblage du servomoteur SG90 avec la carte Arduino UNO se fait à l'aide de 3 broches (FigII.8):

- ✓ Rouge → Alimentation 5V.
- ✓ Marron(Noir) → GND.
- ✓ Orange(Jaune) → Signal de positionnement envoyé par l'Arduino.



**Fig.II. 8** Connexion d'un servomoteur SG90 avec la carte Arduino UNO

Pour que la partie hardware de notre lecteur en Braille fonctionne correctement, une application Android est nécessaire. La conception de cette dernière est détaillée dans la partie qui suit.

### II.4. Partie application Android :

Android est un système d'exploitation Open-source destiné au smartphones, tablettes, ordinateurs, objets connectés, etc... Il permet l'interaction entre l'utilisateur et son appareil par le biais de différentes applications. A cet effet, nous Allons décrire l'un des outils permettant la création de l'application Android.

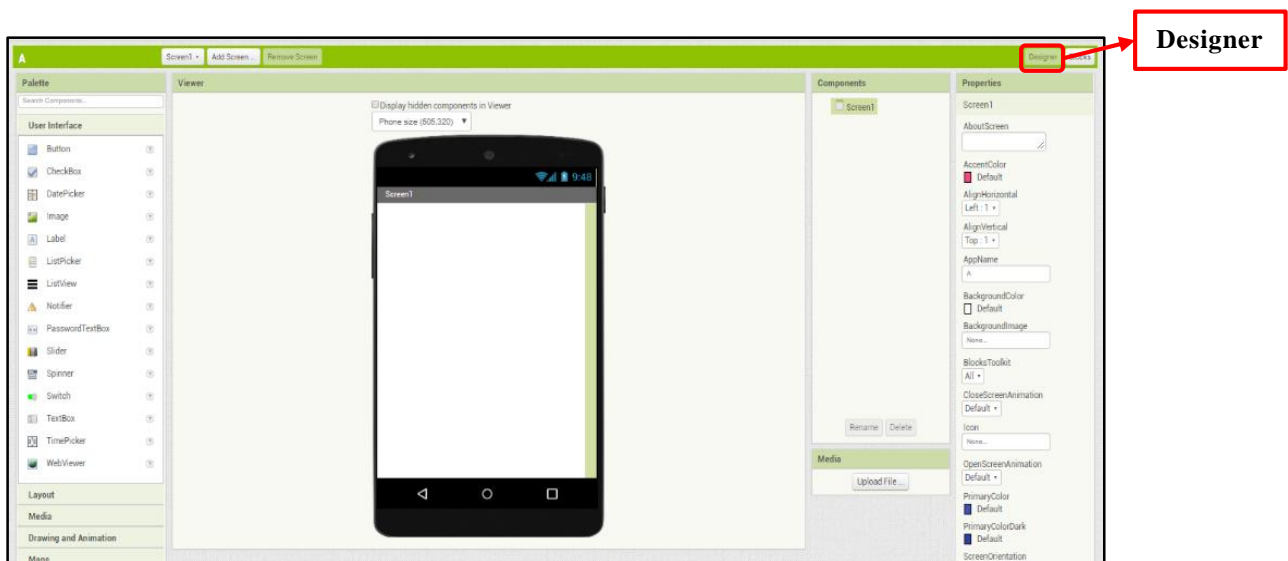
## II.4.1 App Inventor :

App Inventor est un outil de développement en ligne pour les téléphones et les tablettes fonctionnant avec le système d'exploitation Android. C'est une application web à l'origine fournie par Google et actuellement maintenue par MIT (Massachusetts Institute of Technology) [19].

### II.4.1.1 Environnement App Inventor :

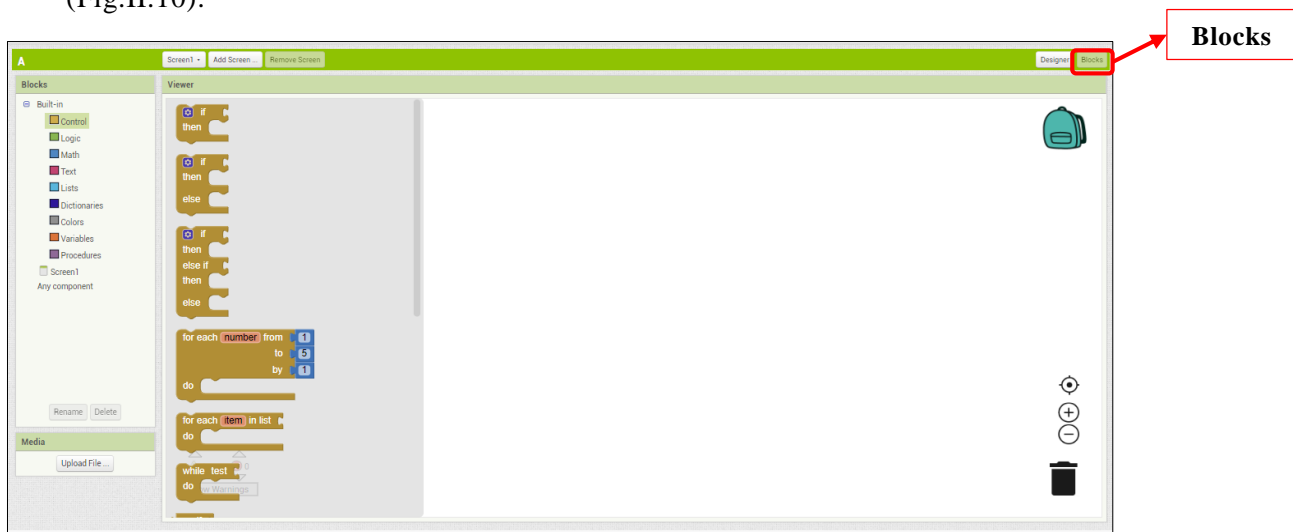
L'environnement App Inventor se compose de deux fenêtres principales qui sont :

- La fenêtre de création « Designer » : Permet de créer l'interface de l'application Android (Fig.II.9).



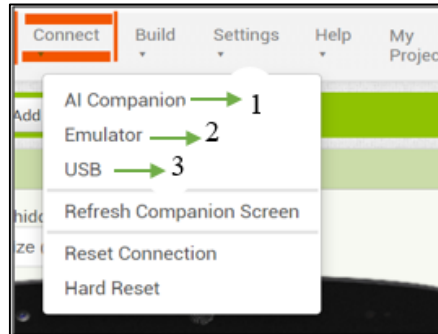
**Fig.II. 9** Fenêtre de création de l'interface de l'application Android sous App Inventor

- La fenêtre de programmation graphique « Blocks » : Permet de programmer l'application (Fig.II.10).



**Fig.II. 10** La fenêtre de programmation sous App Inventor

Une fois la programmation de l'interface est terminée, l'application est sauvegardée, testée et installée sur un smartphone. Le test de l'application peut se faire de trois manières différentes : Compagnon AI, émulateur ou bien par un câble USB. Pour ce faire, il faut d'abord cliquer sur l'onglet « Connect » (Fig.II.11).



**Fig.II. 11** Fenêtre de choix pour le test de l'application Android

➤ **AI Companion :**

Cette option nécessite une installation préalable de « MIT AI2 Companion » sur l'appareil Android (figure II.12.).



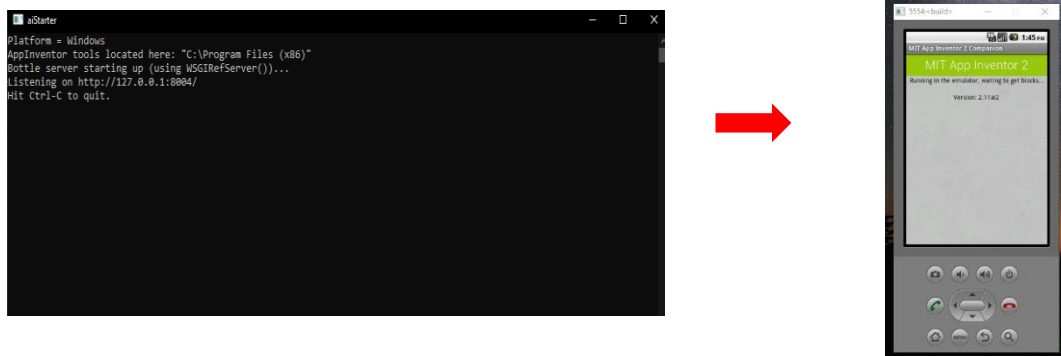
**Fig.II. 12** MIT AI Companion

➤ **USB :**

Cette option permet de tester l'application sur un smartphone connecté à un ordinateur via un câble USB en utilisant le logiciel « aiStarter ».

➤ **L'émulateur :**

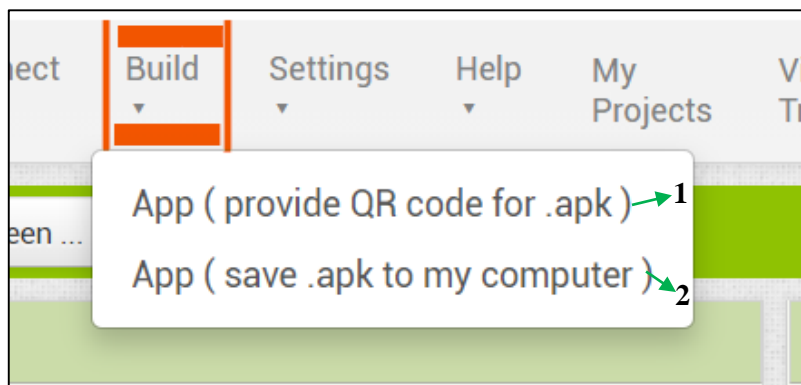
Il permet de tester l'application sur un émulateur Android « aiStarter » installé sur un ordinateur. Une fois aiStarter ouvert, une fenêtre montrant un smartphone virtuel s'affiche (Fig.II.13).



**Fig.II. 13** L'émulateur Android aiStarter

Une fois l'application testée, celle-ci sera installée sur un smartphone. L'installation peut se faire de deux méthodes différentes. Après avoir cliqué sur l'icône « Build » (Fig.II.14) on trouve :

- Construire le QR code de l'application « 1 ».
- Sauvegarder l'application sur l'ordinateur « 2 ».



**Fig.II. 14** Méthodes d'installation de l'application Android

Après avoir détaillé les différents outils (Hardware et application Android), nous présenterons quelques simulations de notre lecteur en Braille faites sous « PROTEUS ISIS ».

## **II.5. Simulation sous « PROTEUS ISIS » :**

Avant de faire la réalisation pratique de notre lecteur en Braille, des tests par simulation sous le logiciel « PROTEUS ISIS » ont été effectués. Nous avons commencé d'abord à simuler la matrice en Braille en utilisant six LED. Ensuite, ces dernières ont été remplacées par des servomoteurs (SG90).

### II.5.1. Présentation du logiciel 'PROTEUS' :

C'est une suite logiciel de conception assistée par ordinateurs (CAO) éditée par la société Labcenter Electronics. C'est un logiciel complet composé de plusieurs outils dont les plus connus sont l'éditeur de schéma électroniques (ISIS) et (ARES) pour le placement et routage des composants. Nous utiliserons dans notre travail uniquement le logiciel ISIS pour simuler le fonctionnement de notre lecteur en Braille. Ceci permettra de détecter et anticiper certaines erreurs et problèmes pouvant surgir en pratique.

### II.5.2. Simulations sous PROTEUS ISIS :

#### II.5.2.1. Simulation avec LED :

L'objectif de cette simulation (Fig II.15) est de disposer six LED sous forme de la matrice de Braille et de les connecter avec la carte Arduino. Ces LED seront allumées ou éteintes selon le caractère en Braille à afficher. Une LED allumée représente un point de la matrice de Braille avec une certaine hauteur, alors qu'une LED éteinte représente un point avec une hauteur nulle. Le programme permettant le fonctionnement de cette simulation est détaillé dans l'annexe 1.

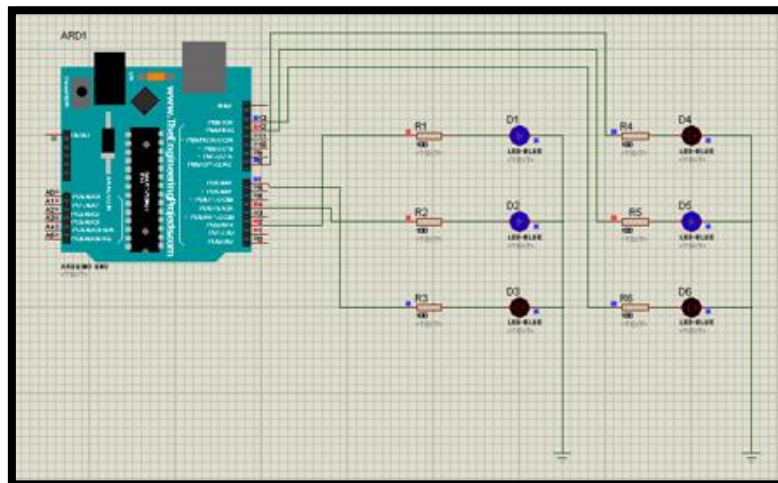
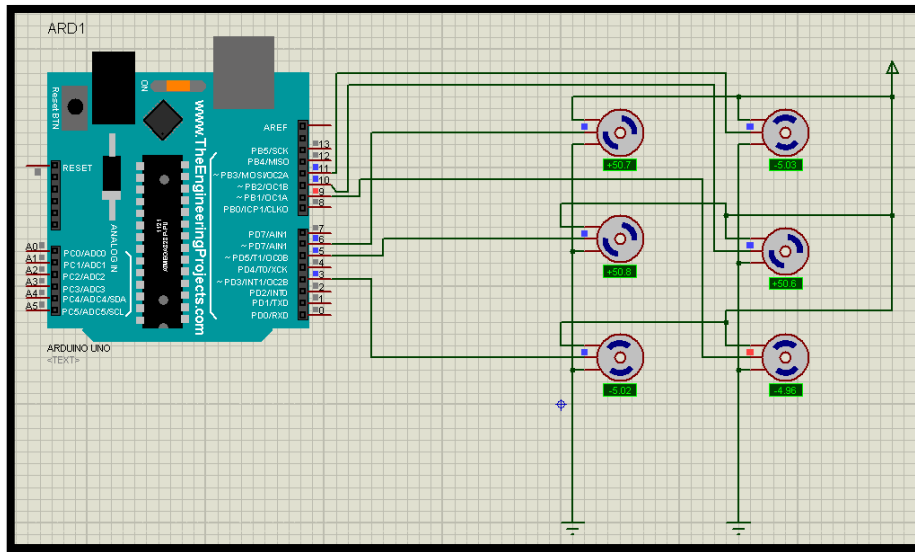


Fig.II. 15 Simulation de la matrice de Braille avec six LED (lettre H en Braille)

#### II.5.2.2. Simulation avec servomoteurs :

Dans ces simulations, les LED sont remplacées par six servomoteurs représentant les six points de la matrice de Braille (Fig.II.16). Le programme de la carte Arduino a aussi été modifié pour s'adapter au servomoteurs SG90 (voir annexe 2). Un servomoteur activé représente un point de la matrice de Braille ayant une certaine hauteur. Un servomoteur désactivé représente un point ayant une hauteur nulle.



**Fig.II. 16** Simulation de la matrice de Braille avec six servomoteurs (lettre H en Braille)

## II.6. Conclusion :

Dans ce chapitre, nous nous sommes focalisés sur la description des éléments nécessaires à la réalisation de notre lecteur en Braille. Nous avons divisé ainsi ces outils en deux parties distinctes et complémentaires. La première se compose d'un ensemble d'outils hardware (carte Arduino, module Bluetooth, LED et servomoteurs SG90). La deuxième concerne l'outils d'édition et de création de l'application Android « MIT APP INVENTOR ». Nous avons par la suite effectué des simulations sous le logiciel Proteus ISIS afin de valider le bon fonctionnement de notre lecteur en Braille. Une première simulation a été réalisée en utilisant six LED représentant les points de la matrice de Braille. Une deuxième simulation a été réalisé en remplaçant ces six LED par six servomoteurs. Le bon fonctionnement de notre lecteur en Braille étant validé par simulation, nous entamerons dans le chapitre suivant la réalisation pratique de ce dernier. Mais avant cela, nous commencerons par définir un seuil de sensibilité chez une personne non-voyante Algérienne.

## **Chapitre III : Réalisation du lecteur en Braille**



### III.1. Introduction :

Le code de Braille est régi par des standards internationaux qui définissent les différentes dimensions relatives aux points de la matrice, leur espacement, leur hauteur ainsi que l'espacement entre deux matrices adjacentes. La hauteur des points de la matrice détermine la capacité d'une personne non-voyante à percevoir un caractère en Braille. Dans ce chapitre, nous commencerons par établir une petite base de données pour déterminer la perception chez les personnes vivantes en Algérie. Ensuite nous entamerons le téléchargement et l'installation de notre application « **BrailleApp** » afin de connecter le smartphone à notre lecteur. Enfin, nous terminerons par la réalisation de notre lecteur en Braille et de le tester d'abord en utilisant des LED avant de passer aux servomoteurs.

### III.2. Etude du sens de toucher en Algérie :

Pour réaliser cette étude, nous avons commencé par fabriquer des pièces contenant des matrices de Braille en utilisant l'imprimante 3D «Printrbot Metal Plus» du laboratoire de Génie Biomédical. Ces matrices ont été modélisées en utilisant le logiciel «Autodesk Inventor» et leur impression 3D a été assurée par le logiciel «Simplify3D». Ensuite, une étude sur le sens de toucher a été menée sur des personnes vivantes dans différentes régions d'Algérie. Ceci nous permettra de définir un seuil de sensibilité lors de l'actionnement des servomoteurs, c'est-à-dire de combien doit être l'angle de rotation du servomoteur pour que les points de la matrice de Braille soient perceptibles (Fig.III.1 – Bloc I).

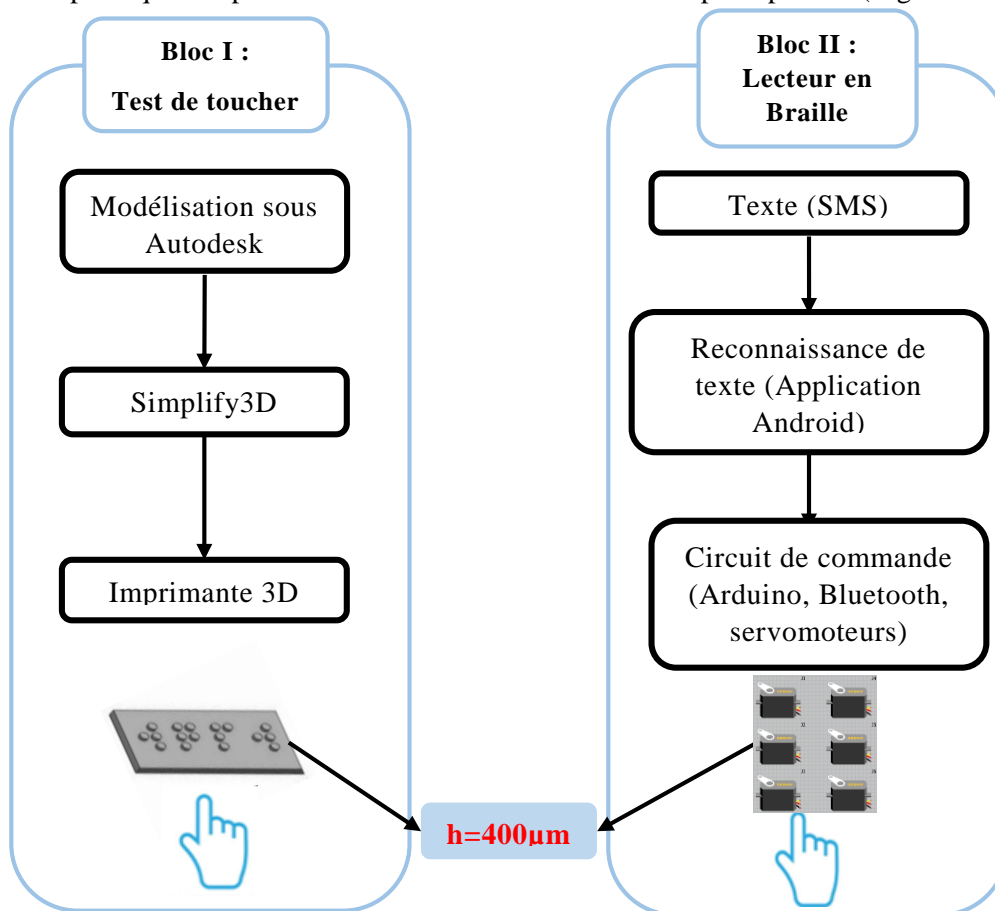
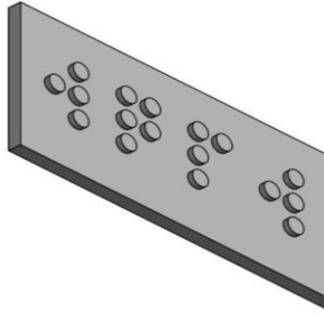


Fig.III. 1 Schéma bloc de la réalisation pratique

### III.2.1. Modélisation sous Autodesk Inventor :

Autodesk Inventor est un logiciel de Conception Assistée par Ordinateur (CAO) développé par la société Autodesk. Il permet de créer et modéliser des objets en 3D grâce à une interface ergonomique et conviviale. Ainsi, en réalisant un dessin 2D et en utilisant l'option « **extrusion** » pour passer à un dessin 3D, afin d'aboutir à des structures comme celle représentée sur la figure III.2. Les Lettres affichée en Braille sur cette dernière sont respectivement de gauche à droite W-Q-P-W.

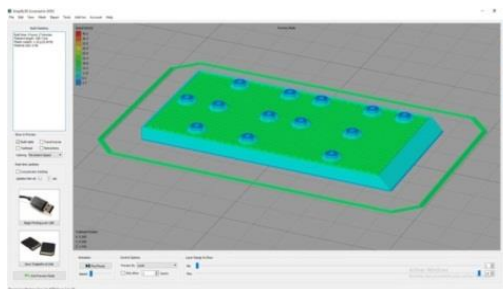


**Fig.III. 2** Structure modélisée sous Autodesk Inventor.

Après avoir terminé la modélisation de la structure, cette dernière est exportée dans un fichier au format 'STL', format utilisé par la majorité des imprimantes 3D.

### III.2.2. Simplify 3D :

C'est un logiciel professionnel destiné à paramétrer et gérer l'impression 3D. Les modèles 3D réalisés et exportés sous le format STL via Autodesk Inventor, sont importés dans le logiciel Simplify3D. Ce dernier va traduire ce fichier STL en une suite d'instruction pour l'imprimante 3D, c'est ce qu'on appelle le G-code. La figure III.3 montre la structure 3D dans la scène d'impression de Simlify3D.



**Fig.III. 3** Scène d'impression du logiciel Simplify3D

### III.2.3. Imprimante 3D :

L'imprimante Printrbot Metal Plus dont dispose le laboratoire de Génie Biomédical GBM est basée sur une technique d'impression dite 'FDM – Fused Deposition Modeling'. Cette technique consiste à chauffer un filament, dans notre cas du plastique PLA (Polylactic acid), en le faisant passer dans une buse d'impression 3D. En se déposant couche par couche sur le plateau d'impression, le PLA refroidit, se durcit et on obtient notre structure imprimée en 3D.

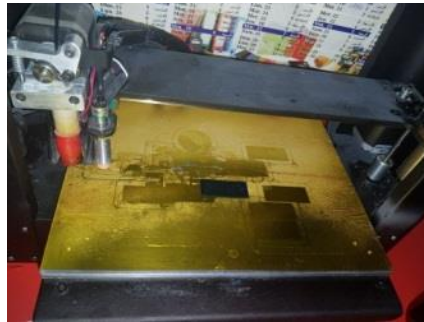


Fig.III. 4 Imprimante 3D Printrbot Metal Plus

### III.2.4. Le test de toucher :

Les structures imprimées contiennent quatre matrices avec des points en relief qui traduisent les lettres W-Q-P-W en Braille. Les dimensions utilisées pour ces matrices sont (Fig.III.5) :

- a (Distance horizontale du centre au centre des points) = 300  $\mu\text{m}$
- b (Distance verticale du centre au centre des points) = 300  $\mu\text{m}$
- c (Espace d'une cellule à une autre) = 600  $\mu\text{m}$
- d (Diamètre de base du point) = 200  $\mu\text{m}$

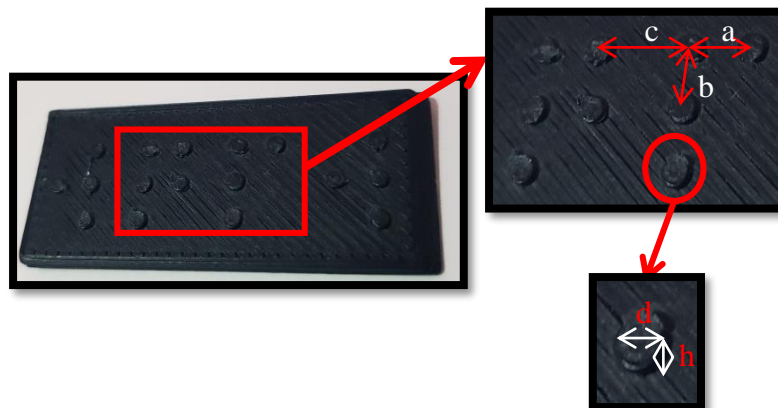


Fig.III. 5 Structures imprimées pour le test du toucher

En réalisant plusieurs structures avec différentes dimensions, nous avons réalisés des tests sur plusieurs personnes afin d'établir la hauteur minimale perceptible. Le tableau de l'annexe 3 dans

l'annexe regroupe l'ensemble de ces tests. D'après les résultats que nous avons obtenus (Annexe 5), la meilleure perception correspond à une hauteur de 400µm. A partir de cette donnée, il s'agira de réaliser un lecteur en Braille utilisant des servomoteurs se déplaçant avec une hauteur de 400µm.

### III.3. Réalisation du lecteur en Braille :

Dans cette partie, nous entamerons la réalisation de notre lecteur en Braille permettant de traduire un texte en code de Braille. En premier lieu, nous commencerons par la création d'une application (BrailleApp) de reconnaissance d'un texte reçu sur smartphone. Par la suite, le code émanant de la reconnaissance de caractères sera envoyé à la carte Arduino via le module HC-05 afin d'actionner les servomoteurs suivant le code en Braille à afficher. Le schéma de fonctionnement de notre lecteur en Braille est représenté dans la figure III.6.

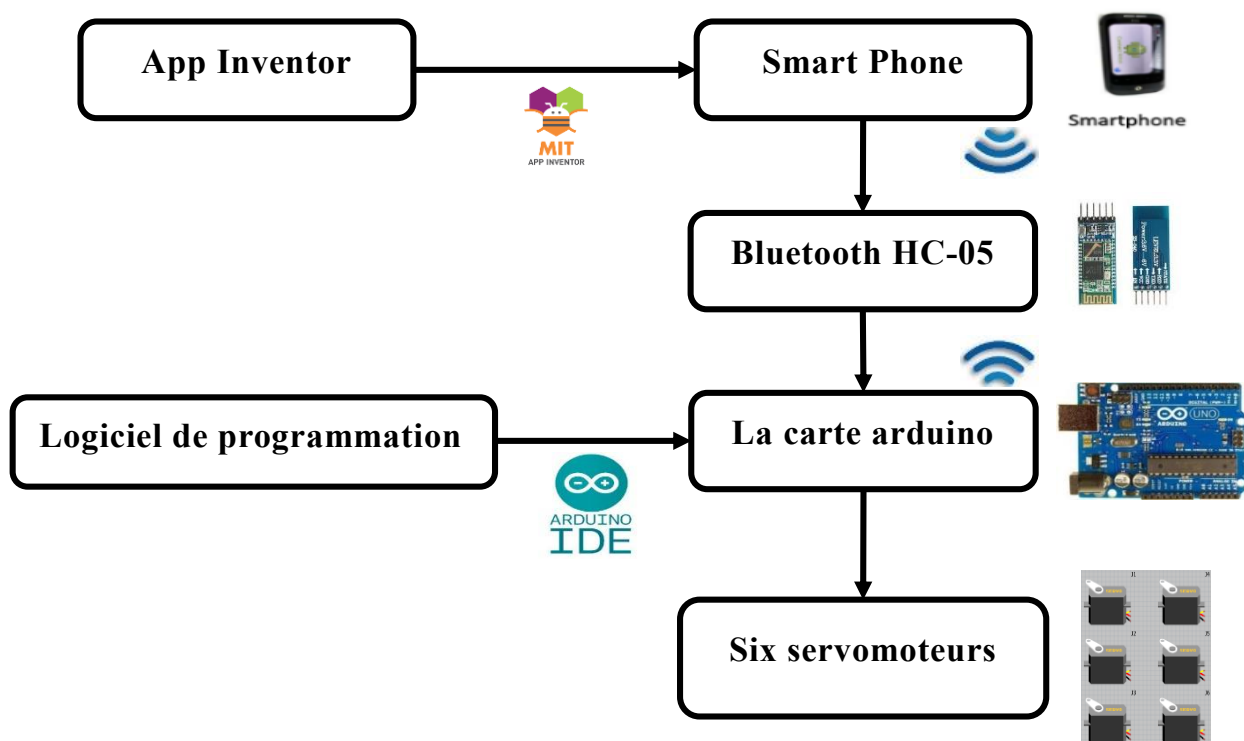


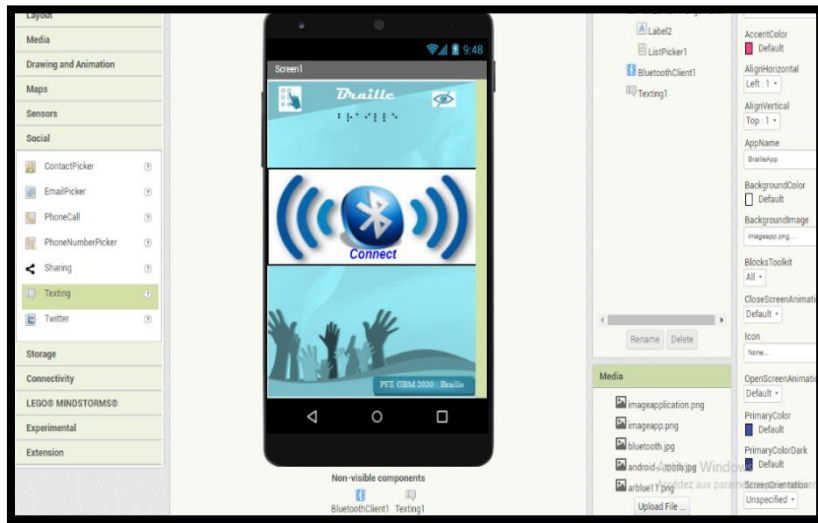
Fig.III. 6 Schéma de fonctionnement du lecteur en braille

#### III.3.1 Application Android :

Comme nous l'avons vu précédemment, nous avons utilisé l'application web « MIT App Inventor » pour créer notre application « **BrailleApp** ». Cette application aura pour but de récupérer un SMS reçu sur un smartphone, le coder et l'envoyer à la carte Arduino via le module Bluetooth HC-05 pour l'afficher en Braille. Pour créer cette application nous avons procédé comme suit :

Nous avons d'abord commencé par créer l'interface de notre application en utilisant l'outil « designer » de MIT App Inventor. Notre application Android contient un bouton « Connect » pour

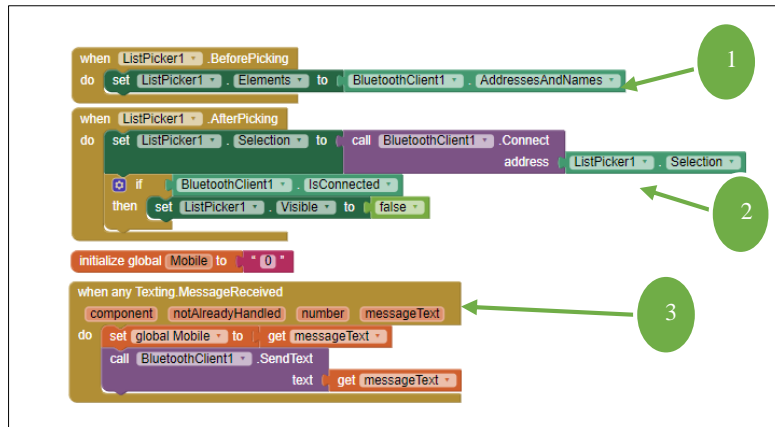
connecter le module Bluetooth HC-05 au smartphone à l'aide de l'outil « BluetoothClient1 » (Fig.III.7). Un autre outil « Texting » est utilisé pour permettre l'identification des SMS.



**Fig.III. 7** L'interface de l'application Android créée

Après avoir créé notre interface, nous entamons la partie programmation des blocs comme suit :

1. Récupération de la liste des objets connectables au Bluetooth.
2. Connexion Bluetooth.
3. Réception d'un SMS.



**Fig.III. 8** Programmation de l'application Android

Après avoir réalisé l'interface de l'application et la programmation des blocs, l'application sera testée en utilisant l'option « AI Companion » (Fig.III.9).

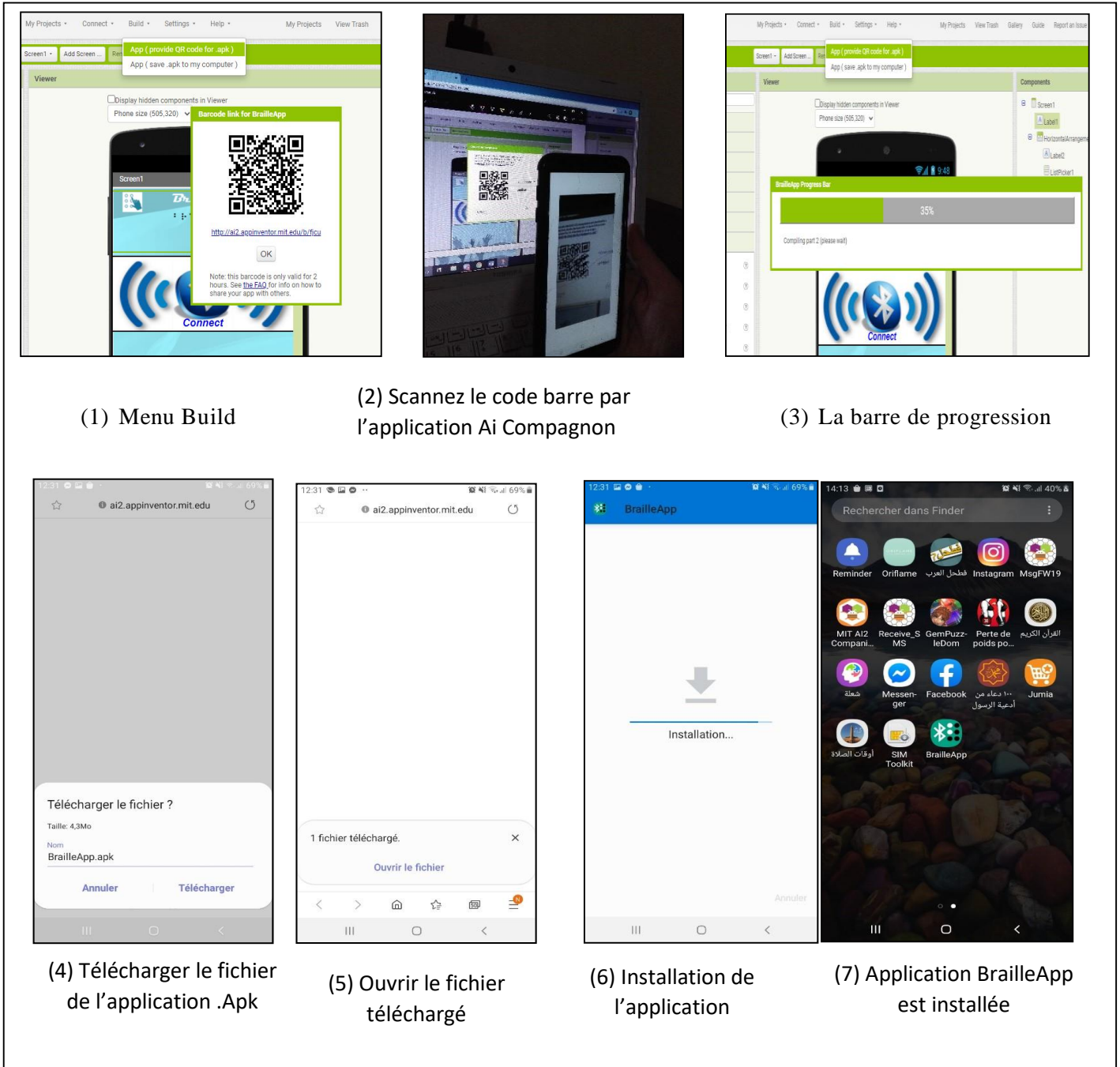
Nous commencerons d'abord par cliquer sur l'onglet « connect » de l'application web MIT App Inventor, on sélectionne ensuite AI Companion (Fig.III.9-(1)). Une fenêtre s'affiche ensuite avec un code barre (Fig.III.9-(2)). L'application AI Companion sur le smartphone est ensuite lancée (figure III.9-(3)). Nous n'aurons plus alors qu'à scanner l'application (figure III.9-(4)) et attendre que l'application s'installe (figure III.9-(5)). Une fois installée, cette dernière est lancée et le test du lecteur en Braille peut commencer (figure III.9-(6)).



**Fig.III. 9** Etapes pour tester et installer l'application "BrailleApp"

On peut aussi télécharger et installer l'application sur le smartphone en générant un fichier APK (Fig.III.10-(1)). Cette application est téléchargée ensuite dans le smartphone en scannant le QR code (Fig.III.10-(2)) et il ne restera plus alors qu'à cliquer sur le fichier APK (Fig.III.10-(5)) pour lancer l'installation (Fig.III.10(6)).





**Fig.III. 10** L'installation de l'application en utilisant le QR code.

Après avoir lancé l'application sur le smartphone, la fenêtre principale de l'application s'ouvre et attend l'action sur le bouton « connect ». En appuyant sur ce dernier, une liste de périphérique Bluetooth est affichée, il faut alors sélectionner le périphérique Bluetooth HC-05 (Fig.III.11).



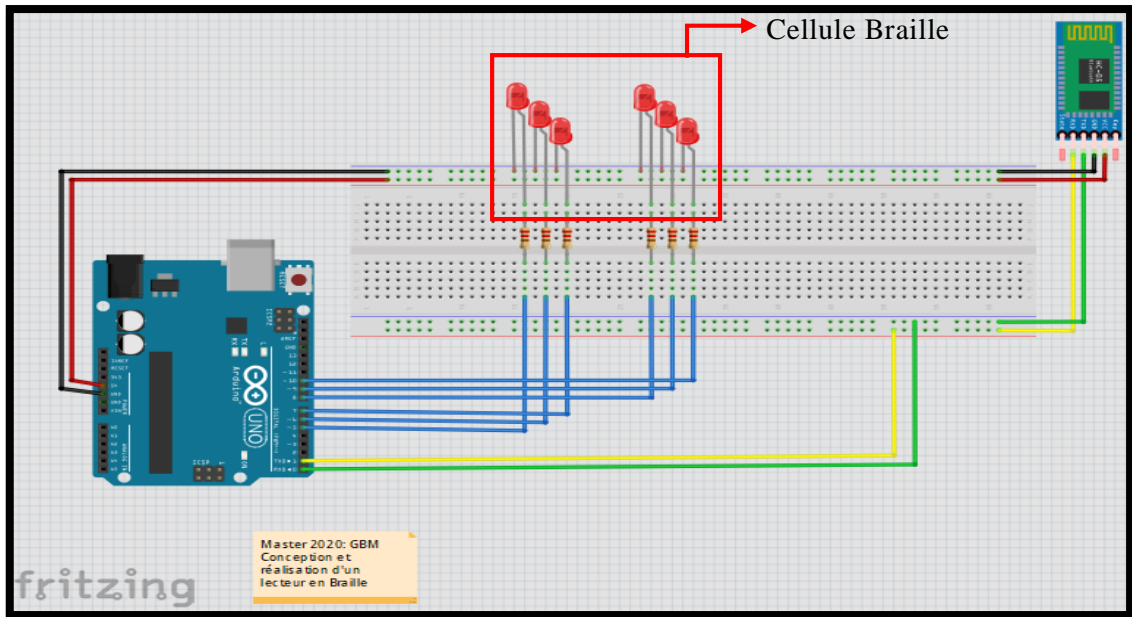
**Fig.III. 11** Connexion Bluetooth entre smartphone et module HC-05

Pour tester l'application pratiquement, nous commencerons tout d'abord par un test avec six LED placés sous forme d'une matrice de Braille.

### III.3.2. Lecteur en Braille en utilisant des LED :

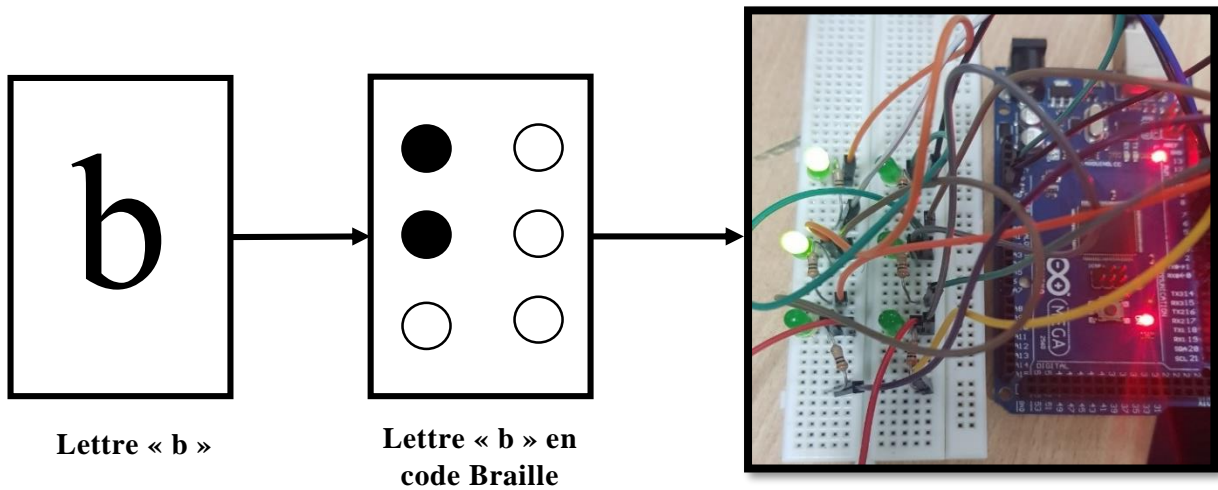
Sur une plaque d'essais, nous avons branché la carte Arduino, le module HC-05 et des LED disposées sur trois lignes et deux colonnes comme le montre la figure III.12.





**Fig.III. 12** Schéma du lecteur en Braille utilisant des LED

En recevant un SMS sur notre smartphone, l'application Android va reconnaître le texte et le transférer vers la carte Arduino en utilisant le programme de l'annexe 6. La figure.III.13 montre le code de Braille correspondant à la lettre « b ».



**Fig.III. 13** Affichage de la lettre "b" en Braille

Dans le cas d'un SMS, on ne parle plus d'une seule lettre à afficher, mais bien d'une chaîne de caractère, le programme doit alors être modifié en conséquence (annexe 4). Le tableau 3 montre un exemple d'un texte reçu en SMS et sa transcription en code de Braille en temps réel.

<b>Message reçu</b>	<b>Merci</b>				
<b>Code de braille associé</b>					
<b>L'affichage du SMS reçu dans les LEDs</b>					
<b>SMS reçu</b>	<b>labo gbm</b>				
<b>Code de braille correspond</b>					
<b>L'affichage du SMS</b>	<p style="text-align: center;"><b>l</b></p>	<p style="text-align: center;"><b>a</b></p>	<p style="text-align: center;"><b>b</b></p>	<p style="text-align: center;"><b>o</b></p>	<p style="text-align: center;"><b>espace</b></p>
	<p style="text-align: center;"><b>g</b></p>	<p style="text-align: center;"><b>b</b></p>	<p style="text-align: center;"><b>m</b></p>		

**Tableau 3.** L'affichage des deux mots 'merci' et 'labo gbm' en code de Braille respectivement sur notre lecteur en Braille.

Le bon fonctionnement de de notre lecteur en Braille en utilisant des LED nous amène à l'étape de remplacement de ces dernières par les servomoteurs SG90.

### III.3.3. Lecteur en Braille en utilisant des servomoteurs :

La carte Arduino sera reprogrammée afin d'envoyer des impulsions vers les servomoteurs SG90 suivant l'angle que l'on souhaite obtenir (déplacement) pour respecter le seuil de sensibilité établie précédemment ( $400\mu\text{m}$ ) (voir annexe 5).

A priori, des tests doivent être menés pour calibrer et trouver le bon angle permettant d'atteindre les  $400\mu\text{m}$  souhaité. Ainsi, un travail sur l'emplacement des servomoteurs doit être étudié afin de respecter les espacements entre les points des matrices et les matrices adjacentes. Malheureusement, tous ces tests n'ont pas pu être menés par manque de composants et compte tenu les conditions sanitaires de cette année à cause du COVID-19.

Néanmoins, le bon fonctionnement du lecteur en Braille en utilisant un seul servomoteur (seul disponible) a été vérifié. Le câblage du lecteur en Braille final est représenté dans la figure III.14.

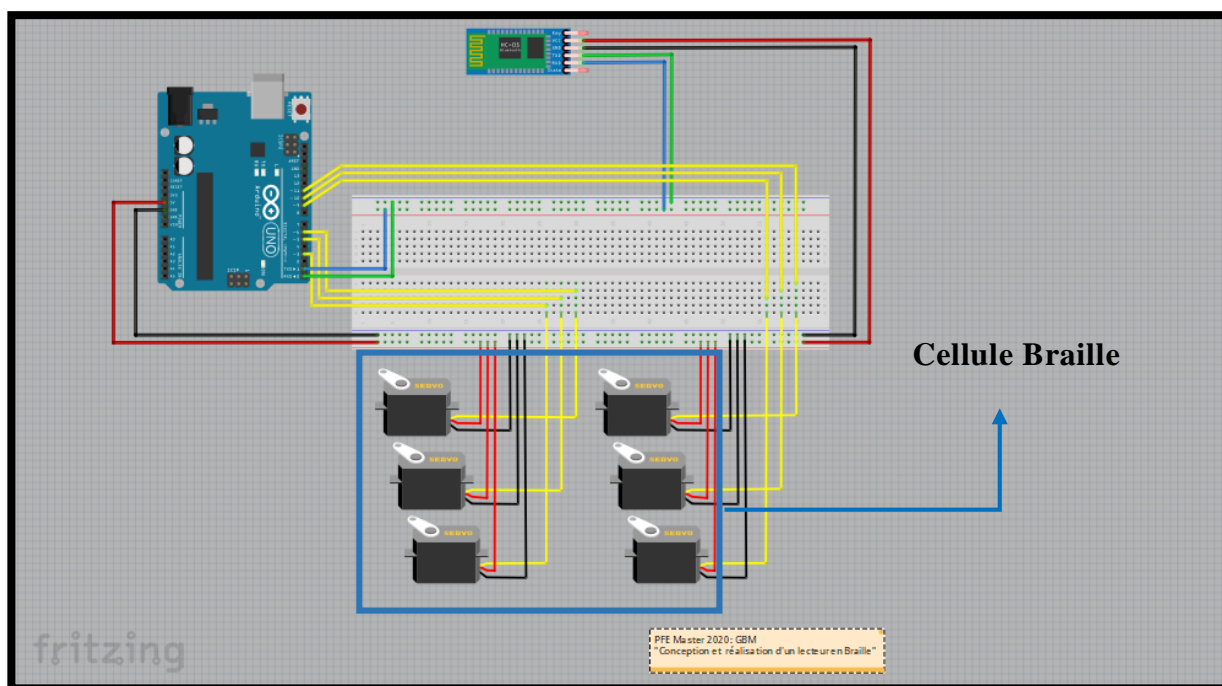


Fig.III. 14 Schéma du lecteur en Braille utilisant des servomoteurs

Après avoir assuré le bon fonctionnement des servomoteurs sous PROTEUS, notre idée était de venir placer six servomoteurs dans un boîtier et les disposer de manière à respecter l'agencement d'une matrice de Braille. En actionnant par la suite ces servomoteurs, des tiges placées sur ces derniers vont sortir du Boîtier (selon le code de Braille) en se déplaçant de  $400\mu\text{m}$ . Enfin, la personne en contact avec notre lecteur en Braille pourra lire le texte convertis et affiché en code de Braille.

### **III.4. Conclusion :**

Dans ce chapitre nous avons présenté les différentes étapes de la réalisation pratique de notre lecteur en Braille. La première a consisté à établir un seuil de sensibilité chez des personnes vivant en Algérie où nous avons pu établir à partir des résultats obtenus un seuil de 400 $\mu$ m. L'application Android « **BrailleApp** » a ensuite été réalisé et installé sur un smartphone ainsi que la partie hardware comportant la carte Arduino, module HC-05, LED et servomoteurs qui a été réalisée. Un premier test concluant en utilisant des LED a permis de vérifier la bonne marche de notre lecteur, ce qui nous a permis de passer à un test avec des servomoteurs. Malheureusement, tous les tests n'ont pas pu être menés, soit par manque de composants, soit à cause des conditions sanitaires exceptionnelles de cette année. Cela n'empêche en rien le fait que ces tests peuvent être considérés comme perspectives futures avec des optimisations plus poussées de ce lecteur en Braille.

## **Conclusion générale et perspectives**

## **Conclusion générale et perspectives :**

Le cadre de ce projet découle de l'intérêt porté sur la conception d'un système de conversion d'un SMS reçu sur un smartphone en code braille à l'aide d'un lecteur en braille. Ce dernier est dédié aux personnes non-voyantes, pour leur faciliter de lire un texte. Notre travail a été procédé selon trois volets principaux afin de réaliser un lecteur en Braille complet :

Le premier volet de notre mémoire a été entamé par un aperçu général sur l'intérêt de braille pour les personnes non-voyantes et à la création d'une petite base de données dans notre région afin de déterminer la meilleure perception chez les algériens.

Le second volet est porté sur la réalisation d'un circuit électronique contenant différents éléments tels que : Une plaque d'essais, un module Bluetooth HC-05, une carte Arduino UNO, six LED et six servomoteurs SG90. Notre circuit électronique consiste à commander les LED et/ou les Servomoteur selon le code de Braille à afficher. Nous rappelons que ces LED et/ou servomoteur sont positionnés sous forme d'une matrice de Braille et que l'utilisation des LED été juste pour assurer le bon fonctionnement de notre circuit afin de ne pas griller les servomoteurs.

Le troisième volet est pour but de monter une application Android qui se communique avec notre circuit électronique via le module Bluetooth. L'application permet d'émettre les SMS reçus dans le smartphone vers la carte Arduino. Cette dernière permet, à partir les déclarations dans le programme de la carte Arduino IDE, de donner les ordres pour activer ou désactiver les servomoteurs afin d'afficher les caractères reçus en code de Braille.

Après avoir achevé les trois volets cités, nous pouvons dire que nous avons consolider nos connaissances en électronique et en informatique et que ce travail réalisé nous a été grandement profitable. Cependant, Il est à noter que nous nous sommes confrontés à plusieurs problèmes dans la réalisation pratique en raison de la crise sanitaire du Coronavirus que traverse actuellement le pays. En dépit de ces difficultés, les résultats obtenus à travers cette étude, qu'ils soient pratiques ou théoriques, sont des résultats satisfaisants.

En perspective, nous pensons à remplacer les servomoteurs par des actionneurs qui nous pourrions faire les surélever avec un hauteur contrôlable et perceptible ( $400\mu\text{m}$ ) afin de concevoir un lecteur portatif et facile à utiliser. Il consiste aussi une amélioration dans l'application Android.

Enfin, Nous souhaitons vivement que ce projet puisse servir comme élément de base pour d'autres études plus approfondies.

## Références :

- [1] « OMS | Jusqu'à 45 millions d'aveugles dans le monde et ces chiffres sont en augmentation », *WHO*. <https://www.who.int/mediacentre/news/releases/2003/pr73/fr/> (consulté le sept. 18, 2020).
- [2] « Journée mondiale de la vue : 180 000 non-voyants en Algérie », *Djazair*. <https://www.djazair.com/fr/horizons/14670> (consulté le févr. 01, 2020).
- [3] A. Sabiani, « Louis BRAILLE », *Rev. Francoph. Orthopt.*, vol. 8, n° 3, p. 234-237, juill. 2015, doi: 10.1016/j.rfo.2015.08.002.
- [4] « Notre histoire », *association Valentin Haüy*, juill. 13, 2016. <https://www.avh.asso.fr/fr/l'association/notre-histoire> (consulté le janv. 22, 2020).
- [5] « rocbo : orthographe, phonétique, sms > Charles Barbier de La Serre - Sonographie ». <https://rocbo.lautre.net/orthog/sonographie.html> (consulté le janv. 22, 2020).
- [6] « Signalisation Braille pour PMR - Gerner Signalisation ». <https://www.gerner-signalisation.com/categories/1-accessibilite-pmr/1-braille> (consulté le janv. 30, 2020).
- [7] « Petit Prince Collection - Livres en braille ». <https://petit-prince-collection.com/lang/brailles.php?lang=fr> (consulté le janv. 30, 2020).
- [8] « Marburg Medium Braille Font Standard – PharmaBraille ». <https://www.pharmabraille.com/pharmaceutical-braille/marburg-medium-font-standard/> (consulté le sept. 18, 2020).
- [9] « Top-Braille, lauréat du concours Lépine, convertit du texte imprimé en braille et en sons », déc. 12, 2019. <https://www.01net.com/actualites/top-braille-laureat-du-concours-lepine-convertit-du-texte-imprime-en-braille-et-en-sons-516483.html?page=2> (consulté le déc. 12, 2019).
- [10] « Plages braille, saisie et lecture électronique du braille pour aveugles », déc. 12, 2019. <https://www.cecias.com/technologie-braille-relief/terminal-braille/plage-braille.html> (consulté le déc. 12, 2019).
- [11] C. Russey, « The Dot Braille Smartwatch Lets the Visually Impaired People See the World from a Different Angle », *Wearable Technologies*, oct. 19, 2018. <https://www.wearable-technologies.com/2018/10/the-dot-braille-smartwatch-lets-the-visually-impaired-people-see-the-world-from-a-different-angle/> (consulté le janv. 30, 2020).
- [12] M. Hussein, M. S. Yaacob, A. R. Musa, et M. Y. Abdullah, « Development of Piezoelectric Braille Cell Control System Using Microcontroller Unit (MCU) », mars 2017, Consulté le: mars 02, 2017. [En ligne]. Disponible sur: <https://pdfs.semanticscholar.org/2bc1/17e77a830be374a527b2a35a4f43f704653e.pdf>.
- [13] « Conception et réalisation d'un micro-actionneur à forte amplitude de déflexion ». .
- [14] *BrailleBox: Android Things Braille news display*. 2017.
- [15] X. Wu, S. H. Kim, H. Zhu, C. H. Ji, et M. G. Allen, « A refreshable braille cell based on pneumatic microbubble actuators », *J Microelectromechanical Syst*, vol. 21, n° 4, p. 908–916, 2012, doi: 10.1109/JMEMS.2012.2190043.
- [16] Y. A. Badamasi, « The working principle of an Arduino », in *2014 11th International Conference on Electronics, Computer and Computation (ICECCO)*, Abuja, Nigeria, sept. 2014, p. 1-4, doi: 10.1109/ICECCO.2014.6997578.
- [17] V. K. Singh, A. Sahu, A. Beg, B. Khan, et S. Kumar, « Speed & Direction Control of DC Motor through Bluetooth HC-05 Using Arduino », in *2018 International Conference on Advanced Computation and Telecommunication (ICACAT)*, Bhopal, India, déc. 2018, p. 1-3, doi: 10.1109/ICACAT.2018.8933698.
- [18] M. S. Mounir, « Conception et réalisation d'un bras manipulateur a 3 degré de liberté », p. 87.
- [19] D. SAIDI, « Conception et réalisation d'un bateau miniature piloté par androïde ». 07 2018, Consulté le: mars 03, 2020. [En ligne]. Disponible sur: <https://dl.ummo.dz/handle/ummo/6454>.

## Annexes :

### Annexe 1. Programme Arduino pour la simulation d'allumage de six LED :

```
char data;
void setup(){
pinMode(2,OUTPUT); pinMode(4,OUTPUT); pinMode(7,OUTPUT); pinMode(8,OUTPUT);
pinMode(12,OUTPUT); pinMode(13,OUTPUT);
Serial.begin(9600);
}
void loop(){
data = Serial.read();
{
// H
digitalWrite(2,HIGH); digitalWrite(4,HIGH); digitalWrite(7,LOW);
digitalWrite(8,LOW); digitalWrite(12,HIGH); digitalWrite(13,LOW);
delay(2000);
// E
digitalWrite(2,HIGH); digitalWrite(4,LOW); digitalWrite(7,LOW);
digitalWrite(8,LOW); digitalWrite(12,HIGH); digitalWrite(13,LOW);
delay(2000);
// L
digitalWrite(2,HIGH); digitalWrite(4,HIGH); digitalWrite(7,HIGH);
digitalWrite(8,LOW); digitalWrite(12,LOW); digitalWrite(13,LOW);
delay(2000);
// L
digitalWrite(2,HIGH); digitalWrite(4,HIGH); digitalWrite(7,HIGH);
digitalWrite(8,LOW); digitalWrite(12,LOW); digitalWrite(13,LOW);
delay(2000);
// O
digitalWrite(2,HIGH); digitalWrite(4,LOW); digitalWrite(7,HIGH);
digitalWrite(8,LOW); digitalWrite(12,HIGH); digitalWrite(13,LOW);
delay(2000);
}
}
```



## Annexe 2. Programme Arduino pour la simulation de fonctionnement de six servomoteurs :

```
int pinMoteur;
void setup(){
pinMode(2,OUTPUT); pinMode(4,OUTPUT); pinMode(7,OUTPUT); pinMode(8,OUTPUT);
pinMode(12,OUTPUT); pinMode(13,OUTPUT);
Serial.begin(9600);
}
void loop(){
pinMoteur = Serial.read();
{
// H
digitalWrite(2,HIGH); digitalWrite(4,HIGH); digitalWrite(7,LOW);
digitalWrite(8,LOW); digitalWrite(12,HIGH); digitalWrite(13,LOW);
delay(3000);
// E
digitalWrite(2,HIGH); digitalWrite(4,LOW); digitalWrite(7,LOW);
digitalWrite(8,LOW); digitalWrite(12,HIGH); digitalWrite(13,LOW);
delay(3000);
// L
digitalWrite(2,HIGH); digitalWrite(4,HIGH); digitalWrite(7,HIGH);
digitalWrite(8,LOW); digitalWrite(12,LOW); digitalWrite(13,LOW);
delay(3000);
// L
digitalWrite(2,HIGH); digitalWrite(4,HIGH); digitalWrite(7,HIGH);
digitalWrite(8,LOW); digitalWrite(12,LOW); digitalWrite(13,LOW);
delay(3000);
// O
digitalWrite(2,HIGH); digitalWrite(4,LOW); digitalWrite(7,HIGH);
digitalWrite(8,LOW); digitalWrite(12,HIGH); digitalWrite(13,LOW);
delay(3000);
}}
```

### Annexe 3. Base de données :

	Sexe	Age	Willaya	Fonction	300µm	400µm	500µm	600µm
1	F	23	Tlemcen	Etudiante		*		
2	F	19	Ain Timouchent	Etudiante	*			
3	F	18	Saida	Etudiante			*	
4	H	42	Djelfa	Mécanicien			*	
5	F	18	Chlef	Etudiante	*			
6	F	18	Chlef	Etudiante		*		
7	H	25	Tlemcen	Plombier		*		
8	F	18	Tebessa	Etudiante		*		
9	F	21	Tindouf	Etudiante		*		
10	F	21	Maghnia	Femme au Foyer		*		
11	F	22	Mascara	Etudiante			*	
12	F	22	Djelfa	Etudiante			*	
13	H	22	ELBayadh	Etudiant		*		
14	F	20	Mecheria	Etudiante			*	
15	F	22	Mecheria	Etudiante			*	
16	F	21	Mecheria	Etudiante		*		
17	H	23	Chlef	Etudiant		*		
18	F	22	Bechar	Etudiante		*		
19	F	23	Bechar	Etudiante		*		
20	H	62	Maghnia	Agriculteur		*		
21	H	33	Maghnia	Commerçant		*		
22	H	29	Maghnia	Commerçant		*		
23	H	31	Maghnia	Ingénieur		*		
24	F	27	Maghnia	Secrétaire		*		
25	F	19	Adrar	Etudiante				*
26	H	56	Maghnia	Commerçant		*		
27	H	26	Maghnia	Commerçant		*		
28	F	11	Maghnia	Elève	*			

## Annexe 4. Programme Arduino de premier test (Application + circuit de commande avec des LED)

```
char f;
int led1=5;int led2=6;int led3=7;int led4=8;int led5=9;int led6=10;
void setup()

{ pinMode(led1,OUTPUT);pinMode(led2,OUTPUT);pinMode(led3,OUTPUT);pinMode(led4,OUTPUT);pinMode(led5, OUTPUT);pinMode(led6, OUTPUT); Serial.begin(38400);
  (Serial.available(>0);}
void loop() {
  {
  f= Serial.read();
  Serial.println(f);
  }
  // a
  if(f=='a'||f=='A')
  {
  digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
  digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
  delay(1000);
  digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
  digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
  delay(500);
  }
  // b
  else if(f=='b'||f=='B')
  {
  digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
  digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
  delay(1000);
  digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
  digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
  delay(1000);
  }
  // c
```

```

else if(f=='c' || f=='C')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// d
else if(f=='d' || f=='D')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// e
else if (f=='e' || f=='E')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// f

else if(f=='f' || f=='F')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
}

```

```

delay(1000);
}
// g

else if(f=='g'||f=='G')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// h

else if (f== 'h'||f=='H')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// i

else if(f=='i'||f=='I')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(8,HIGH); digitalWrite(12,LOW); digitalWrite(13,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// j

else if(f=='j'||f=='J')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);

```

```

delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// k
else if(f=='k'||f=='K')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}

//l
else if (f=='l'||f=='L')
{
digitalWrite(led1,HIGH);digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}

else if (f=='m'||f=='M')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}

else if (f=='n'||f=='N')

```

```

{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
else if (f=='o' || f=='O')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// p
else if(f=='p' || f=='P')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// q
else if(f=='q' || f=='Q')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// r

```

```

else if(f=='r'||f=='R')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// s
else if(f=='s'||f=='S')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// t
else if(f=='t'||f=='T')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// u
else if(f=='u'||f=='U')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,HIGH);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}

```



```

}
// v
else if(f=='v'||f=='V')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,HIGH);
delay (1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// w
else if(f=='w'||f=='W')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,HIGH);
delay (1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// x
else if(f=='x'||f=='X')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,HIGH);
delay (1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// y
else if(f=='y'||f=='Y')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,HIGH);
delay (1000);
}

```

```

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// z
else if(f=='z'||f=='Z')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,HIGH);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay (1000);
}
// 1
if(f=='1')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 2
else if(f=='2')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 3
else if(f=='3')
{

```

```

digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 4
else if(f=='4')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 5
else if (f=='5')
{
digitalWrite(led1,HIGH); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 6
else if(f=='6')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 7

```

```

else if(f=='7')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 8
else if (f== '8')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 9
else if(f=='9')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(8,HIGH); digitalWrite(12,LOW); digitalWrite(13,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// 0
else if(f=='0')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay (1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}

```

```

}
// .
else if(f=='.')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,HIGH);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// ,
else if(f==',')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// ;
else if(f==';')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// :
else if(f==':')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);
}

```

```

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// /
else if(f=='/')
{
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// ?
else if(f=='?')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,HIGH);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// !
else if(f=='!')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// @
else if(f=='@')
{

```

```

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);

delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);

delay(1000);
}
// #
else if(f=='#')
{
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,HIGH);

delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);

delay(1000);
}
// +
else if(f=='+')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);

delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);

delay(1000);
}
// -
else if(f=='-')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);

delay(1000);

digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);

delay(1000);
}
// *

```

```

else if(f=='*')
{
digitalWrite(2,LOW); digitalWrite(4,LOW); digitalWrite(7,HIGH); digitalWrite(8,LOW);
digitalWrite(12,HIGH); digitalWrite(13,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// <
else if(f=='<')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,HIGH);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// >
else if(f=='>')
{
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,LOW);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// (
else if(f=='(')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,HIGH);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}

```



```

}
// )
else if(f=='')
{
digitalWrite(led1,LOW); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,HIGH); digitalWrite(led6,HIGH);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// _
else if(f=='_')
{
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,HIGH);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,HIGH);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// and
else if(f=='&')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,LOW); digitalWrite(led6,HIGH);
delay(1000);
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);
}
// espace
else if(f==' ')
{
digitalWrite(led1,HIGH); digitalWrite(led2,HIGH); digitalWrite(led3,HIGH);
digitalWrite(led4,HIGH); digitalWrite(led5,HIGH); digitalWrite(led6,HIGH);
delay(1000);
}

```

```
digitalWrite(led1,LOW); digitalWrite(led2,LOW); digitalWrite(led3,LOW);
digitalWrite(led4,LOW); digitalWrite(led5,LOW); digitalWrite(led6,LOW);
delay(1000);  }}
```

### **Annexe 5. Programme Arduino de deuxieme test (Application + circuit de commande avec des servomoteurs) :**

```
#include <Servo.h>
char f;

Servo servo;Servo servo1;Servo servo2;Servo servo3;Servo servo4;Servo servo5;
void setup()
    {
        servo.attach(6);
servo1.attach(5);servo2.attach(3);servo3.attach(11);servo4.attach(10);servo5.attach(9);
servo.write(90);servo1.write(90);servo2.write(90);servo3.write(90);servo4.write(90); servo5.write(90);
        Serial.begin(38400);
        (Serial.available())>0);
    }
void loop() {
    {
        f= Serial.read();
        Serial.println(f);
    }
    // a
    if(f=='a' || f=='A')
    {
servo.write(120);delay(1000);
servo.write(90); delay(500);
    }
    // b
    else if(f=='b' || f=='B')
    {
servo.write(120); servo1.write(120); delay(1000);
```

```

servo.write(90); servo1.write(90); delay(1000);
}
// c
else if(f=='c'||f=='C')
{
servo.write(120); servo3.write(120); delay(1000);
servo.write(90);servo3.write(90); delay(1000);
}
// d
else if(f=='d'||f=='D')
{
servo.write(120); servo3.write(120);servo4.write(120); delay(1000);
servo.write(90);servo3.write(90);servo4.write(90); delay(1000);
}
// e
else if (f=='e'||f=='E')
{
servo.write(120);servo4.write(120); delay(1000);
servo.write(90);servo4.write(90); delay(1000);
}
// f
else if(f=='f'||f=='F')
{
servo.write(120);servo1.write(120);servo3.write(120); delay(1000);
servo.write(90);servo1.write(90);servo3.write(90); delay(1000);
}
// g
else if(f=='g'||f=='G')
{
servo.write(120); servo1.write(120); servo3.write(120); servo4.write(120); delay(1000);
servo.write(90);servo1.write(90);servo3.write(90);servo4.write(90); delay(1000);
}
// h
else if (f== 'h'||f=='H')
{
servo.write(120); servo1.write(120); servo4.write(120); delay(1000);
servo.write(90);servo1.write(90);servo4.write(90); delay(1000);
}
}

```

```

// i
else if(f=='i'||f=='I')
{
servo1.write(120); servo3.write(120); delay (1000);
servo1.write(90);servo3.write(90); delay(1000);
}
// j
else if(f=='j'||f=='J')
{
servo1.write(120);servo3.write(120);servo4.write(120); delay (1000);
servo1.write(90);servo3.write(90);servo4.write(90); delay(1000);
}
// k
else if(f=='k'||f=='K')
{
servo.write(120);servo2.write(120); delay (1000);
servo.write(90);servo2.write(90); delay(1000);
}
//l
else if (f=='l'||f=='L')
{
servo.write(120);servo1.write(120);servo2.write(120); delay(1000);
servo.write(90);servo1.write(90);servo2.write(90); delay(1000);
}
else if (f=='m'||f=='M')
{
servo.write(120);servo2.write(120); servo3.write(120); delay(1000);
servo.write(90);servo2.write(90);servo3.write(90); delay(1000);
}
else if (f=='n'||f=='N')
{
servo.write(120);servo2.write(120); servo3.write(120);servo4.write(120); delay(1000);
servo.write(90);servo2.write(90);servo3.write(90);servo4.write(90); delay(1000);
}
else if (f=='o'||f=='O')
{
servo.write(120);servo2.write(120); servo4.write(120); delay (1000);
servo.write(90);servo2.write(90);servo4.write(90); delay (1000);
}

```

```

}

// p
else if(f=='p'||f=='P')
{
servo.write(120);servo1.write(120);servo2.write(120);servo3.write(120); delay (1000);
servo.write(90);servo1.write(90);servo2.write(90);servo3.write(90); delay (1000);
}

// q
else if(f=='q'||f=='Q')
{
servo.write(120);servo1.write(120);servo2.write(120);servo3.write(120);servo4.write(120);
delay (1000);
servo.write(90);servo1.write(90);servo2.write(90);servo3.write(90);servo4.write(90);
delay (1000);
}

// r
else if(f=='r'||f=='R')
{
servo.write(120); servo1.write(120); servo2.write(120); servo4.write(120); delay (1000);
servo.write(90);servo1.write(90);servo2.write(90);servo4.write(90); delay (1000);
}

// s
else if(f=='s'||f=='S')
{
servo1.write(120); servo2.write(120);servo3.write(120); delay (1000);
servo1.write(90);servo2.write(90);servo3.write(90); delay (1000);
}

// t
else if(f=='t'||f=='T')
{
servo1.write(120);servo2.write(120); servo3.write(120);servo4.write(120); delay (1000);
servo1.write(90);servo2.write(90);servo3.write(90);servo4.write(90); delay (1000);
}

// u
else if(f=='u'||f=='U')
{
servo.write(120);;servo2.write(120);servo5.write(120); delay (1000);
}

```

```

servo.write(90);servo2.write(90);servo5.write(90); delay (1000);
}
// v
else if(f=='v'||f=='V')
{
servo.write(120);servo1.write(120);servo2.write(120);servo5.write(120); delay (1000);
servo.write(90);servo1.write(90);servo2.write(90);servo5.write(90); delay (1000);
}
// w
else if(f=='w'||f=='W')
{
servo1.write(120);servo3.write(120);servo4.write(120);servo5.write(120); delay (1000);
servo1.write(90);servo3.write(90);servo4.write(90);servo5.write(90); delay (1000);
}
// x
else if(f=='x'||f=='X')
{
servo.write(120);servo2.write(120);servo3.write(120);servo5.write(120); delay (1000);
servo.write(90);servo2.write(90);servo3.write(90);servo5.write(90); delay (1000);
}
// y
else if(f=='y'||f=='Y')
{
servo.write(120);servo2.write(120);servo3.write(120);servo4.write(120);servo5.write(120);
delay (1000);
servo.write(90);servo2.write(90);servo3.write(90);servo4.write(90);servo5.write(90);
delay (1000);
}
// z
else if(f=='z'||f=='Z')
{
servo.write(120);servo2.write(120); servo4.write(120);servo5.write(120); delay (1000);
servo.write(90);servo2.write(90);servo4.write(90);servo5.write(90); delay (1000);
}
// 1
if(f=='1')
{
servo.write(120); delay(1000);
}

```

```

servo.write(90); delay(1000);
}
// 2
else if(f=='2')
{
servo.write(120); servo1.write(120); delay(1000);
servo.write(90); servo1.write(90); delay(1000);
}
// 3
else if(f=='3')
{
servo.write(120); servo3.write(120); delay(1000);
servo.write(90);servo3.write(90); delay(1000);
}
// 4
else if(f=='4')
{
servo.write(120); servo3.write(120);servo4.write(120); delay(1000);
servo.write(90);servo3.write(90);servo4.write(90); delay(1000);
}
// 5
else if (f=='5')
{
servo.write(120);servo4.write(120); delay(1000);
servo.write(90);servo4.write(90); delay(1000);
}
// 6
else if(f=='6')
{
servo.write(120);servo1.write(120);servo3.write(120); delay(1000);
servo.write(90);servo1.write(90);servo3.write(90); delay(1000);
}
// 7
else if(f=='7')
{
servo.write(120); servo1.write(120); servo3.write(120); servo4.write(120); delay(1000);
servo.write(90);servo1.write(90);servo3.write(90);servo4.write(90); delay(1000);
}

```

```

// 8
else if (f== '8')
{
servo.write(120); servo1.write(120); servo4.write(120); delay(1000);
servo.write(90);servo1.write(90);servo4.write(90); delay(1000);
}
// 9
else if(f=='9')
{
servo1.write(120); servo3.write(120); delay (1000);
servo1.write(90);servo3.write(90); delay(1000);
}
// 0
else if(f=='0')
{
servo1.write(120);servo3.write(120);servo4.write(120); delay (1000);
servo1.write(90);servo3.write(90);servo4.write(90); delay(1000);
}
// .
else if(f=='.')
{
servo1.write(120);servo4.write(120);servo5.write(120); delay(1000);
servo1.write(90);servo4.write(90);servo5.write(90); delay(1000);
}
// ,
else if(f==',')
{
servo1.write(120); delay(1000);
servo1.write(90); delay(1000);
}
// ;
else if(f==';')
{
servo1.write(120);servo2.write(120); delay(1000);
servo1.write(90);servo2.write(90); delay(1000);
}
// :
else if(f==':')

```



```

{
servo1.write(120);servo4.write(120); delay(1000);
servo1.write(90);servo4.write(90); delay(1000);
}
// /

else if(f=='/')
{
servo2.write(120);servo3.write(120); delay(1000);
servo2.write(90);servo3.write(90); delay(1000);
}
// ?

else if(f=='?')
{
servo1.write(120);servo2.write(120);servo5.write(120); delay(1000);
servo1.write(90);servo2.write(90);servo5.write(90); delay(1000);
}
// !

else if(f=='!')
{
servo1.write(120); servo2.write(120); servo4.write(120); delay(1000);
servo1.write(90); servo2.write(90);servo4.write(90); delay(1000);
}

// @

else if(f=='@')
{
servo2.write(120); servo3.write(120); servo4.write(120); delay(1000);
servo2.write(90); servo3.write(90); servo4.write(90); delay(1000);
}
// #

else if(f=='#')
{
servo2.write(120); servo3.write(120); servo4.write(120); servo5.write(120); delay(1000);
servo2.write(90); servo3.write(90); servo4.write(90); servo5.write(90); delay(1000);
}
// +

```

```

else if(f=='+')
{
servo1.write(120); servo2.write(120);servo4.write(120); delay(1000);
servo1.write(90); servo2.write(90);servo4.write(90); delay(1000);
}
// -
else if(f=='-')
{
servo1.write(120);servo4.write(120); delay(1000);
servo1.write(90);servo4.write(90); delay(1000);
}
// *
else if(f=='*')
{
servo2.write(120);servo4.write(120); delay(1000);
servo2.write(90);servo4.write(90); delay(1000);
}
// <
else if(f=='<')
{
servo.write(120); servo1.write(120);servo5.write(120); delay(1000);
servo.write(90); servo1.write(90);servo5.write(90); delay(1000);
}
// >
else if(f=='>')
{
servo2.write(120); servo3.write(120); servo4.write(120); delay(1000);
servo2.write(90); servo3.write(90); servo4.write(90); delay(1000);
}
// (
else if(f=='(')
{
servo1.write(120); servo2.write(120);servo4.write(120); servo5.write(120); delay(1000);
servo1.write(90); servo2.write(90);servo4.write(90); servo5.write(90); delay(1000);
}
// )
else if(f=='')
{

```

```

servo1.write(120); servo2.write(120);servo4.write(120); servo5.write(120); delay(1000);
servo1.write(90); servo2.write(90);servo4.write(90); servo5.write(90); delay(1000);
}
// _
else if(f=='_')
{
servo2.write(120);servo5.write(120); delay(1000);
servo2.write(90);servo5.write(90); delay(1000);
}
// and
else if(f=='&')
{
servo.write(120); servo1.write(120); servo2.write(120); servo3.write(120);servo5.write(120);
delay(1000);
servo.write(90); servo1.write(90); servo2.write(90); servo3.write(90);servo5.write(90);
delay(1000);
}
// espace
else if(f==' ')
{
servo.write(120); servo1.write(120); servo2.write(120); servo3.write(120); servo4.write(120);
servo5.write(120); delay(1000);
servo.write(90); servo1.write(90); servo2.write(90); servo3.write(90); servo4.write(90); servo5.write(90);
delay(1000);
} }

```