

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة الدكتور مولاي الطاهر سعيدة

Université Saida Dr Tahar

Moulay –Faculté de

TECHNOLOGIE



MEMOIRE

Mémoire de fin d'études présenté pour l'obtention du Diplôme de MASTER

En : Automatique

Spécialité : Automatique et Systèmes

Par : Nabi Abderrahmane Rafik et Hachemi Maamar Anouar

Sujet

CONCEPTION ET RÉALISATION D'UN SYSTÈME INTELLIGENT D'ASSISTANCE AUX INDIVIDUS BLESSÉS DANS UN ACCIDENT DE LA ROUTE OU AYANT PERDU CONNAISSANCE.

Soutenue publiquement le **30/06/2024** devant le jury composé de :

Pr. SEKOUR M'hamed

Univ. Saida

Président

Pr. MOSTEFAI Mohammed

Univ. Saida

Rapporteur

Dr. BENMAHDJOUB Mohammed Amin

Univ. Saida

Examineur

Année universitaire 2023/2024



DEDICATION

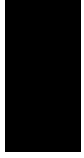
*Dédié avec amour et gratitude à ma famille,
pour leur soutien indéfectible et leurs encouragements.
Merci d'avoir toujours cru en moi.*

*À mes amis,
pour leur amitié sincère et leur présence réconfortante.
Votre soutien m'a été précieux tout au long de ce parcours.*

*À mes professeurs et mentors,
pour leur guidance et leurs précieux conseils.
Votre sagesse et votre expertise ont été des sources d'inspiration inestimables.*

*À tous ceux qui ont contribué, de près ou de loin,
à la réalisation de ce travail.
Votre aide et votre soutien ont été essentiels.
Merci à tous.*

NABI ABDERRAHMANE RAFIK



DEDICATION

*Dédié avec amour et gratitude à ma famille,
pour leur soutien indéfectible et leurs encouragements.
Merci d'avoir toujours cru en moi.*

*À mes amis,
pour leur amitié sincère et leur présence réconfortante.
Votre soutien m'a été précieux tout au long de ce parcours.*

*À mes professeurs et mentors,
pour leur guidance et leurs précieux conseils.
Votre sagesse et votre expertise ont été des sources d'inspiration inestimables.*

*À tous ceux qui ont contribué, de près ou de loin,
à la réalisation de ce travail.
Votre aide et votre soutien ont été essentiels.
Merci à tous.*

HACHEMI MAAMAR ANOUAR



REMERCIEMENT

Je souhaite exprimer ma profonde gratitude à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce travail de mémoire.

Tout d'abord, je remercie sincèrement mon directeur de mémoire, M. MOSTFAI Mohammed, pour sa confiance, son encadrement, ses conseils avisés et son soutien inconditionnel tout au long de cette recherche. Sa rigueur scientifique et son expertise ont été d'une aide précieuse pour mener à bien ce projet.

Je tiens également à exprimer ma reconnaissance à l'ensemble des membres du jury de mémoire pour l'honneur qu'ils me font en acceptant d'évaluer ce travail et pour leurs commentaires constructifs.

Je n'oublie pas de remercier chaleureusement mes collègues et amis de spécialité automatique pour leur soutien moral, leurs discussions enrichissantes et leur convivialité qui ont rendu cette aventure plus agréable.

Une mention spéciale à ma famille pour leur soutien sans faille, leur patience et leurs encouragements constants. Sans leur amour et leur compréhension, la réalisation de ce travail aurait été bien plus difficile.

Je tiens à exprimer ma gratitude au personnel de la protection civile, en particulier au premier lieutenant Abro Mimoune et à l'équipe médicale, pour leur précieuse assistance et leurs conseils pratiques tout au long de cette recherche.

Enfin, je souhaite dédier cette mémoire à Nabi Salim, dont le soutien et l'inspiration m'ont porté tout au long de ce parcours.

Merci à tous.

تتطلب حالات الطوارئ الطبية تدخلاً سريعاً وفعالاً ، حيث تعد كل ثانية ذات أهمية. ومع ذلك ، غالباً ما يتعرض هذه المواقف الحرجة ثغرات في الاتصال بين المرضى وفرق الطوارئ الطبية وخدمات الطوارئ. تهدف هذه الأطروحة إلى معالجة هذه التحديات من خلال اقتراح نظام مساعدة ذكي متضمن.

بعد دراسة سياق خدمات الطوارئ ومبادئ الإسعافات الأولية ، تستكشف الأطروحة منصة أردوينو أونو ومعالجها الدقيق ATmega 328 وتقدم نظرة عامة على إمكاناتهما لتطوير الحلول المتضمنة. ثم يتم تقديم المكونات الإلكترونية المكملية ، مثل جهاز استشعار البصمات ، وشاشة TFT ، ووحدة تخزين.

يفصل جوهر الأطروحة تنفيذ ونمذجة نظام متضمن قادر على التعرف على المرضى وعرض معلوماتهم الطبية وتخزين البيانات ذات الصلة. يهدف هذا النموذج إلى تسهيل تبادل المعلومات الحاسمة أثناء تدخلات الطوارئ.

من خلال الجمع بين مفاهيم الإسعافات الأولية والإلكترونيات المتضمنة وتصميم النظم ، تقدم هذه الأطروحة حلاً مبتكراً لتحسين رعاية المرضى وإنقاذ الأرواح الثمينة.

الكلمات المفتاحية ;

الحالات الطبية الطارئة , المساعدة الطبية , النظام المُدمج ,وحدة التحكم الدقيقة ATmega 328 ,الوصول إلى المعلومات الطبية الحيوية .



Abstract

Medical emergency situations require prompt and effective intervention, where every second counts. However, these critical situations are often hindered by communication gaps between patients, medical teams, and emergency services. This thesis aims to address these challenges by proposing an embedded intelligent assistance system.

After examining the context of emergency services and first aid principles, the thesis explores the Arduino Uno platform and its ATmega328P microcontroller, providing an overview of their capabilities for embedded solution development. Complementary electronic components, such as a fingerprint sensor, TFT display, and storage module, are then presented.

The core of the thesis details the implementation and prototyping of an embedded system capable of identifying patients, displaying their medical information, and storing relevant data. This prototype aims to facilitate the communication of crucial information during emergency interventions.

By combining concepts from first aid, embedded electronics, and system design, this thesis offers an innovative solution to optimize patient care and save precious lives.

Mots clés :

Medical emergencies, Medical assistance, Embedded system, ATmega328P
microcontroller, Access critical medical information



Résumé

Les situations d'urgence médicale nécessitent une intervention rapide et efficace, où chaque seconde compte. Cependant, ces situations critiques sont souvent entravées par des lacunes dans la communication entre les patients, les équipes médicales et les services d'urgence. Cette thèse vise à relever ces défis en proposant un système d'assistance intelligent embarqué.

Après avoir examiné le contexte des services d'urgence et les principes de secourisme, la thèse explore la plateforme Arduino Uno et son microcontrôleur ATmega328P, offrant un aperçu de leurs capacités pour le développement de solutions embarquées. Les composants électroniques complémentaires, tels qu'un capteur d'empreintes digitales, un écran TFT et un module de stockage, sont ensuite présentés.

Le cœur de la thèse détaille la mise en œuvre et le prototypage d'un système embarqué capable d'identifier les patients, d'afficher leurs informations médicales et de stocker des données pertinentes. Ce prototype vise à faciliter la communication d'informations cruciales lors d'interventions d'urgence.

En combinant des concepts de secourisme, d'électronique embarquée et de conception de systèmes, cette thèse apporte une solution novatrice pour optimiser la prise en charge des patients et sauver des vies précieuses.

Mots clés :

Urgences médicales, Assistance médicale, Système embarqué, Microcontrôleur
ATmega328P, Accéder aux informations médicales critiques.



TABLE DES MATIÈRES

LISTE DES FIGURES	xii
LISTE DES TABLEAUX	xiii
LISTE DES ABRÉVIATIONS	xiv
INTRODUCTION GÉNÉRALE	1
1 Interventions médicales d'urgence	3
1.1 Introduction	4
1.2 Service d'Aide Médicale Urgente (SAMU)	4
1.2.1 Organisation et Fonctionnement	5
1.2.2 Missions et responsabilités	5
1.2.3 Innovations technologiques	5
1.2.4 Défis et perspectives	6
1.3 Secourisme	6
1.3.1 DÉFINITION	6
1.3.2 RÔLE DU SECOURISTE	7
1.4 OBLIGATION DE PORTER SECOURS	7
1.5 CLASSIFICATION DES URGENCES	7
1.6 Examen d'un BLESSE. (Bilan et Surveillance)	8
1.6.1 Apprécier l'état de conscience	9
1.6.2 Apprécier la fonction respiratoire	9
1.6.3 Apprécier la fonction circulatoire	10
1.6.4 Surveillance	10
1.7 VICTIME INCONSCIENTE QUI RESPIRE	10
1.7.1 Conduite à tenir	10
1.8 En cas de perte de conscience d'une personne	11
1.8.1 De quelles informations le secouriste a-t-il besoin ?	11

1.8.2	De quelles informations le médecin urgentiste a-t-il besoin ?	11
1.9	Quelle est la différence dans la méthode de traitement entre une personne ordinaire et une personne atteinte d'une maladie chronique lorsqu'elle est inconsciente ?	12
1.10	Conclusion :	13
2	Les microcontrôleurs	14
2.1	Introduction	15
2.2	Définition du microcontrôleur	15
2.3	Différence entre microcontrôleur et microprocesseur	16
2.4	Différent famille des microcontrôleurs :	18
2.5	Etude du l'ATmega 328p :	19
2.5.1	Caractéristique :	19
2.5.2	Structure externe :	21
2.5.3	structure Interne :	23
2.5.4	les modes de faible consommation d'énergie disponibles sur l'atmega328p :	25
2.6	Etude de l'Arduino UNO :	26
2.6.1	Définition du module Arduino :	26
2.6.2	Les gammes de la carte Arduino :	27
2.6.3	Les critères de choix d'Arduino Uno :	33
2.6.4	La constitution de la carte Arduino UNO :	36
2.6.4.1	Partie matérielle :	36
2.6.4.2	Les sources d'alimentation de la carte :	36
2.6.4.3	Les entrées & sorties :	37
2.6.4.4	Les ports de communications :	38
2.7	Descriptions des broches de l'ATmega328P et de la carte Arduino UNO : . . .	39
2.8	la programmation de la carte arduino UNO :	40
2.9	CONCLUSION :	47
3	Composants et matériels utilisés	48
3.1	Introduction	49
3.2	Empreinte digitale :	49
3.2.1	Les caractéristiques de l'empreinte digitale :	49
3.2.2	Capteurs d'empreintes digitales : Un aperçu complet	50
3.2.3	Exploration de la technologie des capteurs :	51
3.2.4	Structure d'un système couplé de reconnaissance d'empreintes digitales : .	52

3.2.5	Étapes clés du processus de reconnaissance d'empreintes digitales : . . .	52
3.2.6	Pourquoi est-ce que l'extraction basée sur des minuties dans la reconnaissance d'empreintes digitales ?	54
3.2.7	Empreintes digitales et minutiae	54
3.3	Capteur d'empreinte FPM10A :	55
3.3.1	Caractéristiques :	56
3.4	Écrans TFT : Introduction et Présentation	57
3.4.1	Introduction aux écrans TFT : Un aperçu	57
3.4.2	Structure des écrans TFT :	58
3.4.3	Fonctionnement des écrans TFT :	59
3.4.4	Types d'écrans TFT :	59
3.4.5	Avantages des écrans TFT :	59
3.4.6	Inconvénients des écrans TFT :	60
3.4.7	Présentation d'un module LCD 2,4 pouces :	60
3.4.8	Fonctionnalités du produit :	61
3.5	Module de carte SD	61
3.5.1	Qu'est-ce qu'un module de carte SD ?	62
3.5.2	Composants d'un module de carte SD	62
3.5.3	Fonctionnement :	63
3.6	Module DS3231 – Horloge en Temps Réel RTC :	63
3.6.1	Caractéristiques et Spécifications :	64
3.6.2	Interface avec les Microcontrôleurs :	64
3.6.3	Applications :	65
3.7	conclusion :	65
4	Partie pratique	67
4.1	Introduction	68
4.2	Description du système	68
4.3	Schéma électrique du montage	70
4.4	Fonctionnements du système	71
4.5	Étude détaillée du système	73
4.5.1	Lecteur d'empreintes :	73
4.5.1.1	Partie du programme lecteur d'empreintes :	74
4.5.2	L'affichage TFT ST7789	75
4.5.2.1	Programme d'affichage :	76
4.5.3	Module de SD Card	76

4.5.3.1	Programme de SD card :	77
4.5.4	L'Horloge RTC DS3231	77
4.5.4.1	Programme de L'Horloge RTC DS3231 :	78
4.5.5	L'alimentation	79
4.5.6	Le microcontrôleur	79
4.6	Fonctionnement du système :	81
4.7	Essais :	89
4.8	CONCLUSION	93
CONCLUSION GÉNÉRALE		94

LISTE DES FIGURES

1.1	une victime inconsciente	4
2.1	brochage typique d'un ATMEGA328 en boîtier PDIP	22
2.2	Schéma interne général de l'ATMEGA328	24
2.3	Architecture interne de l'unité centrale de l'ATMEGA 328	25
2.4	Description Arduino Uno	26
2.5	Arduino Uno Board	27
2.6	Arduino Nano Board	28
2.7	Arduino MEGA Board	28
2.8	Arduino Due Board	29
2.9	Arduino Leonardo Board	29
2.10	30
2.11	Arduino MKR Zero Board	30
2.12	Arduino MKR WiFi 1010 Board	31
2.13	Arduino MKR VIDOR 4000 BOARD	31
2.14	Arduino Lilypad Simple Board	32
2.15	Arduino Lilypad Grande Board	32
2.16	Constitution de la carte Arduino UNO	39
2.17	Schéma simplifié de brochage ATMEGA328P avec la carte Arduino UNO	40
2.18	L'ouverture du logiciel	41
2.19	La fenêtre de l'Arduino IDE.	42
2.20	Le menu Fichier	44
2.21	La barre d'outils.	45
2.22	Paramètre de port de carte.	46
2.23	Paramètre de board de la carte.	46
3.1	Les catégories principales des empreintes digitales [3]	50
3.2	Les principales étapes en images	52
3.3	Un modèle typique d'un modèle d'empreinte digitale	53

3.4	Les minuties	55
3.5	Reconnaissance des empreintes digitales.	55
3.6	Capteur d’empreinte digital.	56
3.7	ECRAN TFT	58
3.8	Structure de l’écran LCD TFT TN	58
3.9	Pin image d’écran.	60
3.10	Module de carte mémoire.	62
3.11	Module DS3231	64
3.12	Interface I2C de DS3231[15]	65
4.1	Schéma bloc du système	69
4.2	L’extérieur de notre appareil	69
4.3	L’extérieur du boîtier de notre appareil	70
4.4	L’intérieur du boîtier de notre appareil	70
4.5	Schéma électrique final sous Proteus	71
4.6	Schéma du module FPM10A sous Proteus	73
4.7	Schéma de l’écran LCD TFT sous Proteus	75
4.8	Schéma de SD-CARD sous Proteus	77
4.9	Schéma de RTC DS3231 sous Proteus	78
4.10	Schéma de batterie avec régulateur sous Proteus	79
4.11	L’organigramme global de microcontrôleur	81
4.12	L’organigramme de la partie d’initialisation du capteur FPM10A	82
4.13	Image scan ready	83
4.14	L’organigramme de la partie d’obtention L’ID d’empreint digitale par FPM10A	84
4.15	Image Try again.	85
4.16	Image found match	86
4.17	L’organigramme de la partie du carte mémoire	87
4.18	Image, information affichée	88
4.19	Image : la date et le temps	89
4.20	Photo de la personne inconsciente avant de scanner ses informations	90
4.22	Photo de l’appareil correspondant trouvé pour la victime	90
4.21	Photo de l’appareil prêt pour le scan	91
4.23	Photo des informations affichées après que la personne correspondante a été trouvée	91
4.24	Photo d’après les informations sur l’heure est affichée	92



LISTE DES TABLEAUX

2.1	Comparaison entre les caractéristiques des microprocesseurs et des microcontrôleurs	17
2.2	Caractéristiques matérielles des cartes Arduino.	34
2.3	Signification des cadres.	43
2.4	Le rôle des boutons des programmations des cartes.	45



LISTE DES ABRÉVIATIONS

Abréviation	Anglais	Français
SAMU	Emergency Medical Aid Service	Service d'Aide Médicale Urgente
L.V.A	Airway Clearance	Libération des Voies Aériennes
P.L.S	Safety Side Position, Recovery Position	Position Latérale de Sécurité
GPIO	General Purpose Input/Output	Entrée/Sortie à Usage Général
ROM	Read-Only Memory	Mémoire Morte
RAM	Random Access Memory	Mémoire Vive
ARM	Advanced RISC Machines	Machines RISC Avancées
SRAM	Static Random-Access Memory	Mémoire Vive Statique
PWM	Pulse Width Modulation	Modulation de Largeur d'Impulsion
AVR	Automatic Voltage Regulator	Régulateur Automatique de Tension
RISC	Reduced Instruction Set Computer	Ordinateur à Jeu d'Instructions Réduit

Abréviation	Anglais	Français
UC	Microcontroller	Microcontrôleur
I2C	Inter-Integrated Circuit, Two-Wire Interface	Circuit Inter-Intégré, Interface à Deux Fils
USB	Universal Serial Bus	Bus Série Universel
EEPROM	Electrically Erasable Programmable Read-Only Memory	Mémoire Morte Programmable Effaçable Électriquement
IDE	Integrated Development Environment	Environnement de Développement Intégré
VIN	Input Voltage	La Tension d'Entrée
SPI	Serial Peripheral Interface	Interface Périphérique Série
LED	Light Emitting Diode	Diode Électroluminescente
RX	Receive	Recevoir
TX	Transmit	Transmettre
TTL	Time To Live, Transistor-Transistor Logic	Le Temps de Vivre, Logique Transistor-Transistor
UART	Universal Asynchronous Receiver Transmitter	Émetteur-Récepteur Asynchrone Universel
FTDI	Future Technology Devices International Limited	Future Technology Devices International Limited
MOSI	Master Out, Slave In	Maître Sortant, Esclave Entrant
MISO	Master In, Slave Out	Maître Entrant, Esclave Sortant

Abbréviation	Anglais	Français
SCK	Serial Clock	Horloge Série
PIC	Programmable Intelligent Computer, Peripheral Interface Controller	Ordinateur Programmable Intelligent, Contrôleur d'Interface Périphérique
SCL	Serial Clock	Horloge Série
SDA	Serial Data	Données Série
DIP	Dual In-Line Package	Ensemble Double En Ligne
CS(SS)	LCD Chip Select Signal, Low Level Enable	Sélection de Puce pour Écran LCD, Activation par Niveau Bas
ALU	32bit Registers for the Arithmetic and Logic Unit	Registres 32bits pour l'Unité Arithmétique et Logique
CAN	Analog-to-Digital Converter	Convertisseur Analogique-Numérique
AREF	Analog Reference Voltage	Tension de Référence Analogique
AVCC	Analog Voltage Common Circuit	Circuit Commun de Tension Analogique
XTAL1	Crystal Oscillator Input	Entrée d'Oscillateur à Quartz
XTAL2	Crystal Oscillator Output	Sortie d'Oscillateur à Quartz
ICSP	Internship and Community Service Programme	Programme de Stages et de Services Communautaires
TFT	Thin-Film Transistor	Transistor à Film Mince

Abréviation	Anglais	Français
FPM 10A	Fingerprint Reader	Lecteur d'Empreintes Digitales
LCD	Liquid Crystal Display	Affichage à Cristaux Liquides
RGB	Red, Green, Blue	Rouge, Vert, Bleu
Super TFT (IPS)	In-Plane Switching	Commutation Dans le Plan
MVA	Multi-domain Vertical Alignment	Alignement Vertical Multi-Domaine
SDIO	Secure Digital Input Output	Entrée/Sortie Numérique Sécurisée
RTC	Real-time Clock	Horloge en Temps Réel
PCB	Printed Circuit Board	Circuit Imprimé



INTRODUCTION GÉNÉRALE

Les situations d'urgence médicale représentent des moments critiques où chaque seconde compte. Une intervention rapide et efficace peut faire la différence entre la vie et la mort. Cependant, ces situations sont souvent entravées par des lacunes dans la communication entre les patients, les équipes médicales et les services d'urgence. Cette thèse vise à examiner ces défis de communication et à proposer des solutions pratiques pour améliorer la gestion des urgences médicales.

Le chapitre 1 expose le contexte des services d'aide médicale urgente, leurs tâches et les principes de base du secourisme. Il souligne les difficultés particulières rencontrées lorsqu'une personne blessée est inconsciente, ne pouvant pas fournir d'informations cruciales sur son état.

Les chapitres suivants explorent différents aspects technologiques visant à relever ces défis. Le chapitre 2 présente la plateforme Arduino Uno et son microcontrôleur ATMEGA328P, offrant un aperçu de leurs capacités et de leur potentiel pour le développement de solutions embarquées. Le chapitre 3 décrit les composants électroniques complémentaires tels qu'un capteur d'empreintes digitales, un écran TFT et un module de stockage, essentiels pour la conception d'un système d'assistance intelligent.

Le chapitre 4 constitue le cœur de cette thèse, détaillant la mise en œuvre et le prototypage d'un système embarqué capable d'identifier les patients, d'afficher leurs informations médicales et de stocker des données pertinentes. Ce prototype vise à faciliter la communication d'informations cruciales lors d'interventions d'urgence.

Enfin, la conclusion générale résumera les principales contributions de cette recherche, ses implications pratiques et les perspectives d'amélioration et de développement futur du système proposé.

En combinant des concepts de secourisme, d'électronique embarquée et de conception de systèmes, cette thèse vise à apporter une solution novatrice pour relever les défis de communication rencontrés dans les situations d'urgence médicale. Elle contribuera ainsi à optimiser la prise en charge des patients et à sauver des vies précieuses.

Chapitre

1

Interventions médicales d'urgence

1.1 Introduction

Pour faire face au temps précieux perdu dans les situations d'urgence médicale, les interventions doivent tenir compte de plusieurs facteurs tels que la communication entre les patients, les équipes médicales et les services d'urgence. Dans la première partie, nous allons exposer le service d'aide médicale urgente, leurs tâches et le secourisme. Dans la partie suivante, nous examinons comment ces lacunes affectent l'efficacité des interventions médicales d'urgence et proposons des solutions pratiques pour une communication efficace dans ces circonstances critiques.



FIGURE 1.1 – une victime inconsciente

1.2 Service d'Aide Médicale Urgente (SAMU)

Le Service d'Aide Médicale Urgente (SAMU) est un élément central du système de santé en France, chargé de la régulation et de la coordination des soins médicaux d'urgence. Connu sous le numéro d'appel d'urgence "15" ou "centre 15", le SAMU joue un rôle crucial dans la gestion des urgences pré-hospitalières, offrant une réponse rapide et adaptée aux situations critiques telles que les accidents, les malaises soudains, et les complications médicales graves [1] [2].

1.2.1 Organisation et Fonctionnement

Le SAMU est structuré autour de centres de réception et de régulation des appels (CRRA), où des médecins régulateurs évaluent les demandes d'aide médicale urgente et orientent les patients vers les services appropriés. Ces centres sont équipés pour gérer une variété de situations, allant des conseils médicaux téléphoniques à l'envoi de véhicules de secours et d'assistance aux victimes (VSAV) ou d'ambulances privées pour des transports sanitaires non médicalisés [3] [4].

1. **Interventions primaires** : Assistance directe sur le lieu de l'incident, stabilisation du patient et transport vers un établissement de soins adéquat.
2. **Interventions secondaires** : Transfert de patients entre établissements hospitaliers nécessitant une surveillance médicale ou des soins intensifs pendant le trajet[5].

1.2.2 Missions et responsabilités

Le SAMU a pour mission de fournir une réponse sanitaire adaptée aux urgences médicales. Cela inclut la gestion des crises sanitaires aiguës et des urgences médicales collectives, telles que les catastrophes naturelles ou les épidémies. Le SAMU est également responsable de déclencher les plans blancs des hôpitaux, qui sont des dispositifs de gestion de crise pour faire face à un afflux massif de victimes[1][6].

1.2.3 Innovations technologiques

Les nouvelles technologies jouent un rôle de plus en plus important dans l'amélioration des services d'urgence. Par exemple, l'intégration de la télémédecine et des dispositifs connectés, comme les lunettes intelligentes, permet une meilleure évaluation et prise en charge des patients à distance. Ces innovations facilitent la communication entre les équipes sur le terrain et les médecins régulateurs, améliorant ainsi l'efficacité des interventions[7][8][9].

1.2.4 Défis et perspectives

Le SAMU fait face à plusieurs défis, notamment la saturation des services d'urgence due à une demande croissante de soins non programmés. Environ 37 % des dossiers de régulation concernent des transports sanitaires non médicalisés, souvent réalisés par des ambulanciers privés. De plus, la régulation médicale doit s'adapter aux nouvelles menaces, telles que les épidémies et les catastrophes naturelles, nécessitant une coordination accrue avec les autres services de secours[3] [4][10].

Le Service d'aide médicale urgente est un pilier essentiel du système de santé français, offrant une réponse rapide et coordonnée aux urgences médicales. Grâce à l'intégration des nouvelles technologies et à une organisation efficace, le SAMU continue d'évoluer pour répondre aux défis actuels et futurs, garantissant ainsi une prise en charge optimale des patients en situation critique.

1.3 Secourisme

1.3.1 DÉFINITION

Ensemble des moyens pratiques et thérapeutiques simples mis en œuvre pour porter secours aux personnes en danger et leur donner les 1ers soins. D'après le Petit Larousse Le secourisme est à la fois une technique et un état d'esprit.[11]

- **Une technique :** Elle s'acquiert et se perfectionne par des exercices pratiques répétés.
- **Un état d'esprit :** un désir de servir, soit seul, soit mieux en équipe, l'action en groupe organisé décuplant l'efficacité.

1.3.2 RÔLE DU SECOURISTE

Le secouriste exerce dans des circonstances diverses, dans la calamité de paix ou de guerre, mais aussi dans la vie de tout le jour. C'est d'abord savoir porter secours efficacement et éviter l'aggravation de l'état d'un accidenté, son rôle [12] :

- Écarte la foule.
- Examine le blessé.
- Pratiquer les gestes d'urgence indispensables.
- Fait prévenir les secours publics.
- Fait transporter ou accompagne le blessé à l'hôpital.

1.4 OBLIGATION DE PORTER SECOURS

Selon le Code pénal, tout individu témoin d'un accident doit signaler l'accident aux secours publics et encourir des peines d'emprisonnement et d'amende. Pour quiconque refuse volontairement de fournir une assistance à une personne en danger sans risque pour lui ou pour des tiers, il pourrait lui prêter, que ce soit pour lui ou pour des tiers, soit en agissant personnellement, soit en lui offrant un secours[13].

1.5 CLASSIFICATION DES URGENCES

Urgence relative, absolue, extrême, pronostic vital engagé. De nombreux termes utilisés par les professionnels des secours sont largement utilisés dans les journaux. Quelles sont les véritables diagnostics médicaux qui se cachent derrière ces expressions, et selon quels critères ?

- **Extrême urgence :**
 - * Les asphyxiés ranimés ou en cours de réanimation.
 - * Les hémorragies « non garrotables »
 - * Les blessés du thorax qui étouffent.

- * Les « grands choqués »

- **Première urgence :**

- * Les porteurs de garrot pour hémorragies artérielles.
- * Les membres écrasés.
- * Les plaies abdominales.
- * Les sujets inconscients, mais respirant spontanément.

- **Deuxième urgence :**

- * Les fractures du crâne conscient, de la colonne vertébrale, du bassin.
- * Les hémorragies arrêtées par pansement compressif.
- * Les plaies profondes, les fractures ouvertes.

- **Troisième urgence :**

- * Les « petits » blessés (petites fractures fermées, plaies peu profondes).

Cette classification n'a rien d'absolu ; le secouriste devra considérer chaque cas en tenant compte des règles générales d'évacuation.[14]

1.6 Examen d'un BLESSE. (Bilan et Surveillance)

L'examen clinique se déroule de la tête aux pieds (haut à gauche) pour repérer les points d'impact visibles par la présence d'hématomes et de contusions. Des contusions, des plaies, des déformations visibles, un saignement qui était au départ inaperçu, un soulèvement asymétrique de la cage thoracique.[14]

Le bilan consiste à vérifier :

- **Le bilan consiste à Vérifier :**

- ★ L'état de conscience.
- ★ La ventilation.
- ★ La circulation.
- ★ Rechercher une éventuelle lésion.

- ★ Hémorragie.
- ★ Plaie,
- ★ Brûlure,
- ★ Fracture.

1.6.1 Apprécier l'état de conscience

Le secouriste doit [15] :

- Poser des questions simples : votre nom ? Ça va ?
- Donner des ordres simples : ouvrez les yeux, serrez-moi la main. . .
- Pincer légèrement la victime (sourd muet).

1.6.2 Apprécier la fonction respiratoire

Le secouriste doit [15] :

- ✓ Assurer la libération des voies aériennes (LVA).
- ✓ Desserrer tout ce qui est serré (cravate, col, ceinture. . .).
- ✓ Ouvrir la bouche de la victime, la nettoyer (enlever bridge, bonbons) et essuyer les mucosités à l'aide d'un linge propre.
- ✓ Placer les quatre doigts d'une main sur le front de la victime.
- ✓ Placer deux doigts de l'autre main sous la pointe du menton en prenant appui sur l'os et non dans la partie molle du menton.
- ✓ Basculer prudemment la tête en arrière.
- ✓ Pencher son oreille et sa joue au-dessus de la bouche et du nez de la victime pour sentir le flux d'air expiré.
- ✓ Observer le soulèvement de la poitrine et du ventre.

1.6.3 Apprécier la fonction circulatoire

- ✓ saignement,
- ✓ coupure de la peau,
- ✓ brûlure,
- ✓ cassure d'un os,[15]

1.6.4 Surveillance

Le secouriste doit jusqu'à l'arrivée des secours spécialisés [15] :

- ✓ continuer à parler à la victime,
- ✓ contrôler régulièrement la fonction ventilatoire,
- ✓ contrôler régulièrement la fonction circulatoire.

1.7 VICTIME INCONSCIENTE QUI RESPIRE

C'est une victime immobile, ne répondant ni aux questions ni aux ordres, mais qui respire[15].

1.7.1 Conduite à tenir

- Le secouriste doit Installer la victime en position latérale de sécurité (P.L.S.), Saisir le poignet du côté du retournement, et étendre doucement le membre supérieur correspondant jusqu'à le mettre légèrement au-delà de la Perpendiculaire à l'axe du corps.
- Se placer à genou, à hauteur des hanches, du même côté, suffisamment éloigné de la victime.
- Placer une main sur l'épaule opposée en fléchissant le coude de la victime (l'avant bras reposant sur celui du secouriste).

- Saisir avec l'autre main la hanche - Retourner ensuite lentement, en gardant les bras tendus, le corps de la victime vers lui, d'un bloc.
- Libérer le membre supérieur de la victime qui reposait sur celui du secouriste et le placer sur le sol en le fléchissant.
- Placer la main qui tenait l'épaule sur la hanche et celle qui se trouvait sur la hanche saisit le mollet, pour fléchir la jambe sur la cuisse, ramener le genou sur le sol et le pied derrière la jambe pour caler la victime.
- Se déplacer et se mettre derrière la tête de la victime afin de compléter éventuellement la bascule de sa tête, et garder la bouche ouverte dirigée vers le sol. - Couvrir la victime - Alerter les secours spécialisés. - Surveiller la victime.[4]

1.8 En cas de perte de conscience d'une personne

En cas de perte de conscience d'une personne quels sont les trois informations à transmettre aux services de secours[15] ?

1.8.1 De quelles informations le secouriste a-t-il besoin ?

- * Vérifier les fonctions vitales (respiration, pouls)
- * Nom complet, âge, maladie en cours de traitement, groupe sanguin et adresse personnelle de la victime.
- * Cela permet au secouriste d'agir en fonction de l'état de la personne.

1.8.2 De quelles informations le médecin urgentiste a-t-il besoin ?

- ✓ Il a besoin des informations susmentionnées, en plus de :
- ✓ La victime souffre-t-elle de maladies chroniques ?
- ✓ Quels médicaments le patient prend-il ?
- ✓ Cela permet au médecin urgentiste de déterminer l'état de santé de la victime

a) **En cas d'hypertension artérielle chez une personne inconsciente, quelle est la méthode de traitement ?**

- * Mesurer la tension artérielle et administrer un médicament par injection.
- * Apaiser, rassurer et réchauffer la victime.

b) **En cas de diabète chez une personne inconsciente, quelle est la méthode de traitement ?**

- * Mesurer la glycémie
- * Si la glycémie est inférieure à 0,6 g/L, donner à la victime quelque chose de sucré
- * Si la glycémie est élevée, le médecin urgentiste administre une injection d'insuline
- * Si aucun appareil de mesure de la glycémie n'est disponible et que le patient prend de l'insuline, lui donner quelque chose de sucré et le transporter à l'hôpital.

1.9 Quelle est la différence dans la méthode de traitement entre une personne ordinaire et une personne atteinte d'une maladie chronique lorsqu'elle est inconsciente ?

a) **Personne ordinaire [15] :**

- * L'histoire médicale n'est pas beaucoup prise en compte.
- * Des examens de base sont effectués pour déterminer la cause de la perte de conscience.
- * L'accent est mis sur le traitement des symptômes aigus.
- * Le patient est surveillé régulièrement jusqu'à ce qu'il reprenne conscience.

b) **Personne atteinte d'une maladie chronique [15] :**

- * L'histoire médicale est un élément essentiel du diagnostic et du traitement (revue des médicaments, état de santé général)
- * Des examens complémentaires sont effectués pour déterminer l'impact de la perte de conscience sur son état de santé (tests de la fonction rénale, hépatique et sanguine)

- * Le traitement peut nécessiter de traiter la cause sous-jacente de la perte de conscience.
- * Une surveillance plus longue peut être nécessaire, surtout si la perte de conscience est due à des complications de son état de santé.
- * Une sensibilisation accrue peut être nécessaire sur la façon d'éviter la perte de conscience, comme la modification du mode de vie ou des médicaments.

c) En cas d'épilepsie chez une personne inconsciente, quelle est la méthode de traitement [15] ?

- * Prendre soin de la personne pour qu'elle ne se blesse pas et l'emmener à l'hôpital (service des urgences)

d) e) En cas d'accident cardiaque chez une personne inconsciente qui prend des médicaments, quelle est la méthode de traitement [15] ?

- La victime doit être emmenée à l'unité de soins intensifs après un transfert rapide à l'hôpital.

1.10 Conclusion :

Ce chapitre aborde les différentes étapes de secours aux personnes accidentées ou blessées et la différence dans la méthode de traitement entre une personne ordinaire et une personne inconsciente. La question se pose plus pour un blessé inconscient, car il n'est pas possible d'interroger. Une solution sera avancée dans le chapitre III, ce qui peut contribuer à résoudre ce problème.

Chapitre

2

Les microcontrôleurs

2.1 Introduction

Le monde de l'électronique embarquée et du prototypage rapide s'est démocratisé ces dernières années grâce à l'émergence de plateformes de développement accessibles et performantes. Parmi les plus populaires, l'Arduino Uno se distingue par sa simplicité d'utilisation, son prix abordable et sa communauté active. Au cœur de cette carte réside le microcontrôleur ATmega328P, un composant puissant et polyvalent qui permet de concrétiser une large gamme de projets.

Ce chapitre explore la plateforme Arduino Uno et se concentre plus particulièrement sur le microcontrôleur ATmega328P qui en est le cœur. Nous commencerons par une introduction à l'Arduino Uno, en examinant ses composants, ses fonctionnalités et ses avantages. Ensuite, nous approfondirons nos connaissances sur l'ATmega328P, en analysant son architecture, ses caractéristiques techniques et ses capacités de traitement.

2.2 Définition du microcontrôleur

Un microcontrôleur, également connu sous les abréviations μC , UC, ou MCU en anglais, est un circuit intégré conçu pour régler une opération spécifique et intégrée dans un système. Il comprend un processeur, une mémoire et des périphériques d'entrée et de sortie sur une seule carte ou une seule puce[16]. Les microcontrôleurs sont utilisés dans de nombreuses industries et applications, telles que la robotique, l'automatisation industrielle, les communications et les appareils médicaux[16].

Les microcontrôleurs sont caractérisés par leur faible consommation électrique et leur coût réduit par rapport aux microprocesseurs polyvalents utilisés dans les ordinateurs personnels[17]. Ils sont fréquemment utilisés dans les systèmes embarqués, comme les contrôleurs des moteurs automobiles, les télécommandes, les appareils de bureau, l'électroménager, les jouets, la téléphonie mobile, etc. [17].

Les microcontrôleurs sont disponibles en différentes gammes, telles que les microcontrôleurs STM32 de STMicroelectronics, les microcontrôleurs PIC et les microcontrôleurs ARM [16].

2.3 Différence entre microcontrôleur et microprocesseur

1) Composition et architecture [18] :

- **Microprocesseur (MPU) :** Un MPU est essentiellement une unité centrale de traitement (CPU) autonome. Il s'agit d'un circuit intégré complexe qui exécute des instructions et effectue des calculs. Les MPU n'ont généralement pas de mémoire ni de périphériques intégrés et s'appuient sur des bus externes pour communiquer avec ces composants. Ils adoptent généralement une architecture de type Von Neumann, où la mémoire programme et la mémoire de données partagent le même espace d'adressage.
- **Microcontrôleur (MCU) :** Un MCU est un système complet sur une puce qui intègre un CPU, de la mémoire (RAM et ROM), des entrées/sorties (E/S) et d'autres composants périphériques sur un seul circuit intégré. Cette architecture compacte permet une communication interne plus rapide et une consommation d'énergie réduite. Les MCU utilisent souvent une architecture Harvard, où la mémoire programme et la mémoire de données sont distinctes, offrant une meilleure efficacité et une meilleure sécurité.

2) Utilisation et fonctionnalités :

- **Microprocesseur[19] :** les MPU sont conçus pour des tâches générales dans les ordinateurs personnels, les serveurs et les stations de travail. Ils offrent une puissance de traitement élevée et un jeu d'instructions vaste et complexe pour gérer des applications exigeantes telles que le traitement d'image, les jeux vidéo et le multimédia.
- **Microcontrôleur[20] :** les MCU sont optimisés pour des tâches spécifiques et dédiées dans des systèmes embarqués, tels que les appareils électroménagers,

les automobiles, les jouets, les dispositifs médicaux et les systèmes de contrôle industriel. Ils offrent une consommation d'énergie réduite, une taille compacte et un coût abordable, ce qui les rend idéaux pour des applications où l'espace, l'autonomie et l'efficacité énergétique sont primordiaux.

3) Avantages et inconvénients[20] :

TABEAU 2.1 – Comparaison entre les caractéristiques des microprocesseurs et des microcontrôleurs

Caractéristique	Microprocesseur	Microcontrôleur
Composition	CPU uniquement	CPU, mémoire, E/S et périphériques sur une puce
Architecture	Von Neumann (généralement)	Harvard (souvent)
Utilisation	Tâches générales (ordinateurs, serveurs)	Tâches spécifiques (systèmes embarqués)
Jeu d'instructions	Vaste et complexe	Restreint et optimisé
Bus externe	Nécessaire	Intégré
Mémoire	Externe	Interne
Périphériques	Externes	Intégrés
Coût	Plus élevé	Moins cher
Complexité	Plus complexe	Plus simple
Consommation d'énergie	Plus élevée	Plus faible
Applications	Ordinateurs, serveurs, stations de travail	Appareils embarqués, systèmes de contrôle

En résumé : Les microprocesseurs et les microcontrôleurs sont des composants électroniques essentiels, mais ils répondent à des besoins différents. Les MPU offrent une puissance de traitement brute pour des tâches générales, tandis que les MCU sont optimisés

pour des applications spécifiques dans des systèmes embarqués, privilégiant l'efficacité, la compacité et le coût. Le choix entre les deux dépend des exigences spécifiques de l'application.

2.4 Différent famille des microcontrôleurs :

Le monde des microcontrôleurs est vaste et diversifié, avec une multitude de familles répondant à des besoins et applications spécifiques. Parmi les plus populaires, on retrouve :

1. **Famille Atmel AVR** : Développée par Atmel (maintenant Microchip), cette famille est reconnue pour sa simplicité d'utilisation, sa faible consommation d'énergie et son large choix de modèles. L'ATmega 328P, utilisé dans l'Arduino Uno, en fait partie [21].
2. **Famille PIC (Microchip)** : Connue pour son architecture RISC performante et sa grande variété de fonctionnalités, la famille PIC est largement utilisée dans les applications industrielles, automobiles et grand public [22].
3. **Famille Intel 8051** : Une référence historique dans le domaine des microcontrôleurs, la famille 8051 est appréciée pour sa robustesse, sa compatibilité et son vaste écosystème d'outils de développement[23].
4. **Famille Motorola 68HC11** : Particulièrement adaptée aux applications de contrôle et d'automatisation, la famille 68HC11 offre un bon équilibre entre performances et prix[24].
5. **Famille ST6 (STMicroelectronics)** : Développée par STMicroelectronics, cette famille se distingue par sa haute performance, sa flexibilité et son intégration de nombreux périphériques[25].
6. **Famille Hitachi H8** : Reconnue pour sa fiabilité et sa consommation d'énergie optimisée, la famille H8 est appréciée dans les applications critiques et les environnements contraignants [21].
7. **Famille C167 (Siemens/Infineon)** : Orientée vers les applications industrielles et de contrôle, la famille C167 offre une architecture puissante et une grande variété de fonctionnalités de sécurité[26].

8. Famille ADuC (Analog Devices) : Spécialement conçue pour les applications de traitement du signal analogique, la famille ADuC intègre des convertisseurs analogique-numérique et numérique-analogique hautes performances [26].

9. Famille PICBASIC (Comfile Technology) : Ciblent les débutants et les amateurs, la famille PICBASIC propose un environnement de développement convivial basé sur le langage BASIC, facilitant l'apprentissage et la programmation des microcontrôleurs PIC[26].

Il est important de noter que chaque famille de microcontrôleurs possède ses propres forces, faiblesses et caractéristiques spécifiques. Le choix de la famille la plus adaptée dépend des besoins et des contraintes de chaque projet.

2.5 Etude du l'ATmega 328p :

L'ATmega 328P est un microcontrôleur 8 bits puissant de la famille AVR, développé par Atmel (maintenant une partie de Microchip Technology). Il combine un riche ensemble de fonctionnalités et de capacités dans un boîtier compact, en faisant un choix populaire pour une large gamme d'applications embarquées.

2.5.1 Caractéristique :

Le microcontrôleur ATMEGA328P est un dispositif CMOS 8 bits de la famille AVR, utilisant une architecture RISC avancée. Voici une fusion des caractéristiques détaillées de ce microcontrôleur, incluant les spécifications supplémentaires fournies [27] :

1. Caractéristiques Générales

- * **Type de Processeur :** AVR 8 bits avec architecture RISC.
- * **Nombre de Broches :** 28 (sur le modèle DIP utilisé en TP).
- * **Package :** DIP 28 broches ou QFN 28 broches.
- * **Fréquence d'Horloge :** 16 MHz (maximum toléré : 20 MHz).

- * **Technologie** :Technologie CMOS haute vitesse pour une consommation d'énergie réduite.

2. Mémoire

- * **Mémoire Flash** :32 ko, programmable par interface série ISP, permettant l'autoprogrammation en système.
- * **EEPROM** :1 Ko, stocke des données non volatiles, même après une mise hors tension.
- * **RAM** :2 Ko, stocke des données volatiles utilisées pendant l'exécution du programme.

3. Entrées/Sorties et Périphériques

- * **Ports d'entrée/sortie (GPIO)** :
 - **PORTB (8 broches)** :Port d'entrée/sortie bidirectionnel.
 - **PORTC (7 broches)** : Port d'entrée/sortie bidirectionnel.
 - **PORTD (8 broches)** :Port d'entrée/sortie bidirectionnel.
- * **Convertisseurs A/N (CAN)** :6 convertisseurs A/N de 10 bits.
- * **Comparateur Analogique** : 1 Comparateur analogique.
- * **Timers** :
 - **1 timer 16 bits (T1)**
 - **2 timers 8 bits (T0, T2).**
 - Gèrent le temps et permettent de générer des signaux de temporisation.
- * **PWM (Pulse Width Modulation)** :6 canaux PWM.
- * **Watchdog** :1 chien de garde (watchdog).

4. Communication et Interfaces

- * **SPI (Serial Peripheral Interface).**
- * **USART (Universal Synchronous/Asynchronous Receiver Transmitter).**
- * **TWI (Two-Wire Interface), équivalent à I²C.**

- * **Interruptions** : 26 interruptions permettant au microcontrôleur de réagir à des événements externes.

5. Alimentation et Conditions de Fonctionnement

- * **Tension d'alimentation** : 8V à 5.5V.
- * **Température de fonctionnement** : -40°C à +85°C.
- * **Broches d'Alimentation** :
 - **VCC (1)** : Alimentation principale du microcontrôleur.
 - **GND (2)** : Masse du circuit.
 - **AVCC (1)** : Broche d'alimentation dédiée aux convertisseurs A/N pour une meilleure précision.
 - **AREF (1)** : Tension de référence externe pour le comparateur analogique.

6. Modes d'Économie d'Énergie

- * **Modes d'Économie d'Énergie** : 5 modes différents pour optimiser la consommation d'énergie.

Ces caractéristiques font de l'ATMEGA328P un choix populaire pour diverses applications, notamment dans les systèmes embarqués et les projets DIY utilisant la plateforme Arduino.

2.5.2 Structure externe :

Le microcontrôleur ATMEGA328P est commercialisé sous plusieurs formes de boîtiers. La forme utilisée dans la carte ARDUINO UNO est un boîtier PDIP 28 broches.[28]

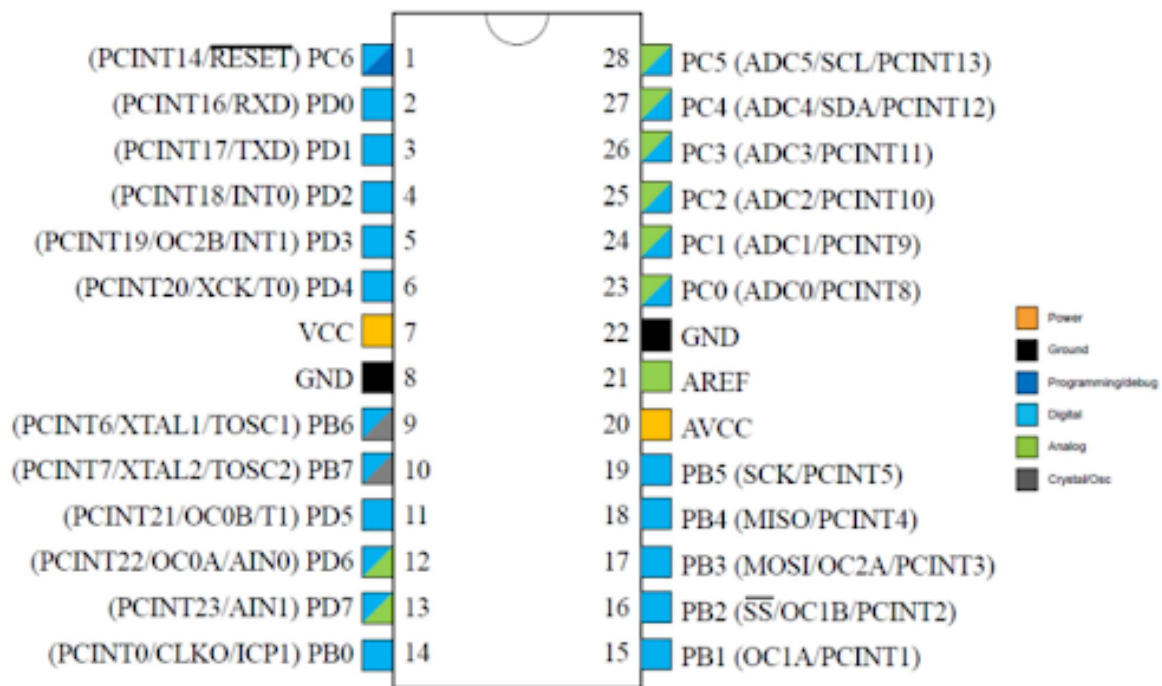


FIGURE 2.1 – brochage typique d'un ATMEGA328 en boîtier PDIP

voici descriptions des broches :

- **VCC** : Tension d'alimentation.
- **GND (Ground)** : masse.
- **Port B (PB7 : PB0)** : Le port B est un port d'entrée-sortie à 8 bits bidirectionnel avec des résistances internes de tirage (pull-up). Il sert aussi de sorties PWM (PB1/OC1 A, PB2/OC1 B, PB3/OC2 A). Selon les réglages de sélection d'horloge ,
- L'oscillateur à quartz externe doit être connecté sur PB6/XTAL1 et PB7/XTAL2 pour cadencer le microcontrôleur.
- **Port C (PC5 : PC0)** : le port C est un port d'entrée-sortie à 7 bits bidirectionnel avec des résistances internes de tirage. Il sert aussi comme entrée du convertisseur A/N (ADC0. ADC5). Les broches PC4/SDA et PC5/SCL sont utilisées par le bus I2C (TWI) s'il est activé. • PC6 : la broche PC6 est utilisée comme entrée RESET.
- **Port D (PD7 : PD0)** : le port D est un port d'entrée-sortie à 8 bits bidirectionnel avec des résistances internes de tirage. Il sert aussi de sorties PWM (PD6/OC0 A, PD5/OC0 B, PD3/OC2 B). Les broches PD2/INT0 et PD3/INT1 peuvent être configurées pour déclencher des interruptions externes.

- Les broches PD0/RXD et PD1/TXD sont utilisées par la communication série USART (RS232) en cas d'activation.
- **AVCC** : c'est une broche de tension d'alimentation pour le convertisseur A/D qui doit être connectée à VCC via un filtre passe-bas pour éviter les parasites.
- **AREF** : est l'entrée de référence analogique pour le convertisseur A/D.

2.5.3 structure Interne :

- **Architecture RISC avancée** : L'ATmega328P est basé sur une architecture RISC (Reduced Instruction Set Computer) avancée, avec 131 instructions puissantes qui s'exécutent généralement en un seul cycle d'horloge. [27].
- **Registres de Travail** : il dispose de 32 registres de travail à usage général, directement connectés à l'Unité Arithmétique et Logique (ALU), permettant d'accéder à deux registres indépendants en une seule instruction [29].
- **Mémoires** : il intègre 32 Ko de mémoire Flash pour le programme, 2 Ko de mémoire SRAM pour les données, et 1 Ko d'EEPROM non volatile [21].
- **Entrées/Sorties** : le microcontrôleur offre 23 lignes d'entrées/sorties programmables, ainsi que des fonctionnalités avancées comme un convertisseur analogique-numérique 10 bits sur 6 canaux, des timers/compteurs flexibles et des interfaces série USART, SPI et I2C. [30].
- **Gestion de l'alimentation** : il dispose de plusieurs modes de gestion de l'alimentation sélectionnables par logiciel, permettant d'optimiser la consommation d'énergie en fonction des besoins de l'application. [31].

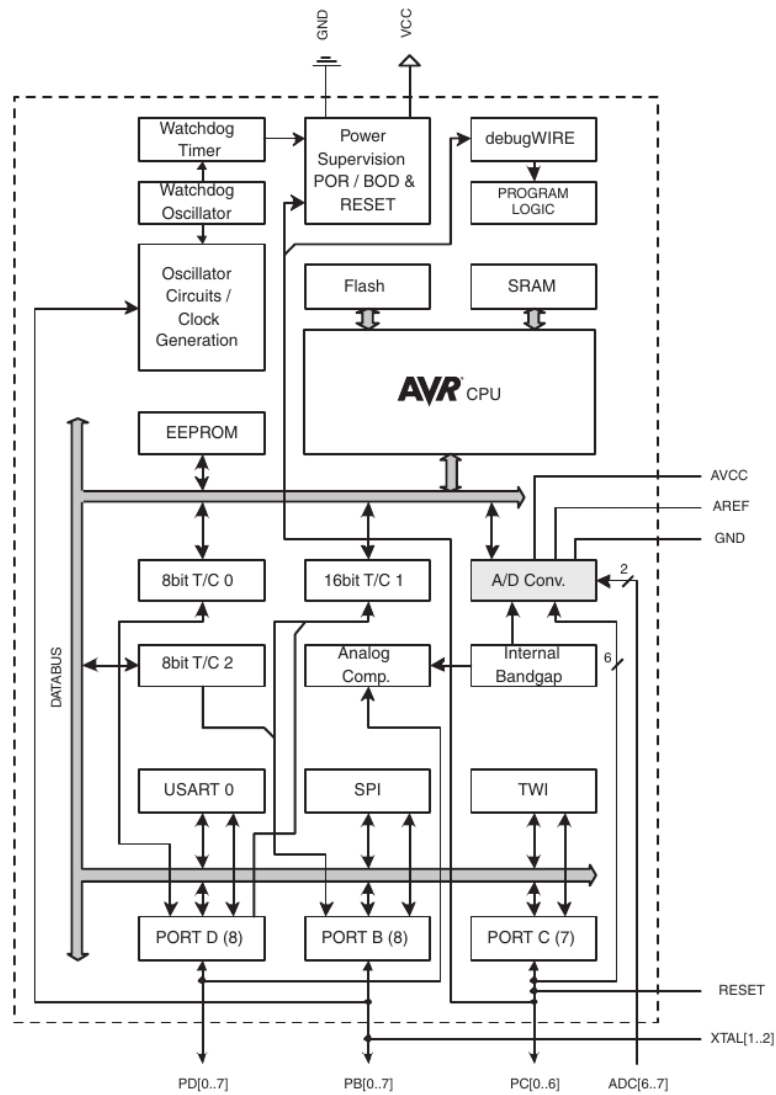


FIGURE 2.2 – Schéma interne général de l'ATMEGA328

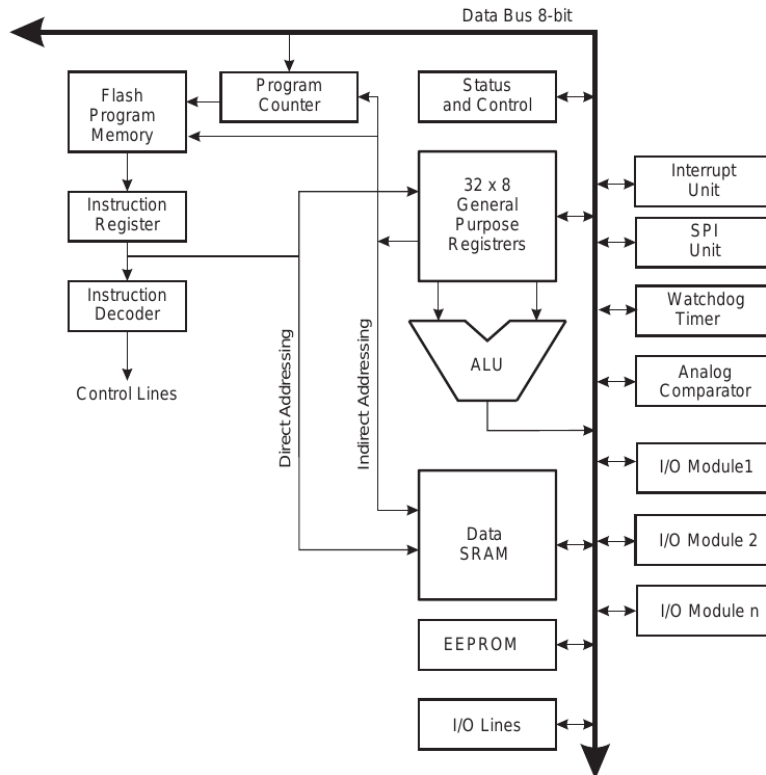


FIGURE 2.3 – Architecture interne de l'unité centrale de l'ATMEGA 328

2.5.4 les modes de faible consommation d'énergie disponibles sur l'atmega328p :

L'ATMEGA328P dispose de plusieurs modes de faible consommation d'énergie sélectionnables par logiciel :

- 1) **Mode Idle** : ce mode permet de mettre le processeur en veille tout en maintenant le fonctionnement des minuteries/compteurs, de l'USART, de l'interface SPI et de l'ADC. La consommation est réduite dans ce mode.
- 2) **Mode Réduction du Bruit ADC** : ce mode permet de réduire le bruit électrique lors de l'utilisation du convertisseur analogique-numérique, ce qui peut être utile pour des applications sensibles au bruit.
- 3) **Mode économie d'énergie** : dans ce mode, le processeur, la plupart des oscillateurs et la plupart des périphériques sont mis en veille, ne laissant actifs que le watchdog timer,

l'interface d'interruption externe et certaines fonctions d'horloge. La consommation est fortement réduite dans ce mode.

- 4) **Mode Mise hors tension** : ce mode met le microcontrôleur complètement hors tension, à l'exception du circuit de détection de remise sous tension. C'est le mode de consommation le plus faible.

Ces différents modes de gestion de l'alimentation permettent aux concepteurs de systèmes à base d'ATMEGA328P d'optimiser la consommation d'énergie en fonction des besoins spécifiques de leur application, contribuant ainsi à une utilisation plus efficace de l'énergie. [32]

2.6 Etude de l'Arduino UNO :

2.6.1 Définition du module Arduino :

L'Arduino Uno est une carte de développement électronique open source basée sur le microcontrôleur ATMEGA328P. Elle offre une interface simple entre le matériel et le logiciel, permettant aux utilisateurs de programmer le microcontrôleur sans connaissances approfondies en électronique[33].

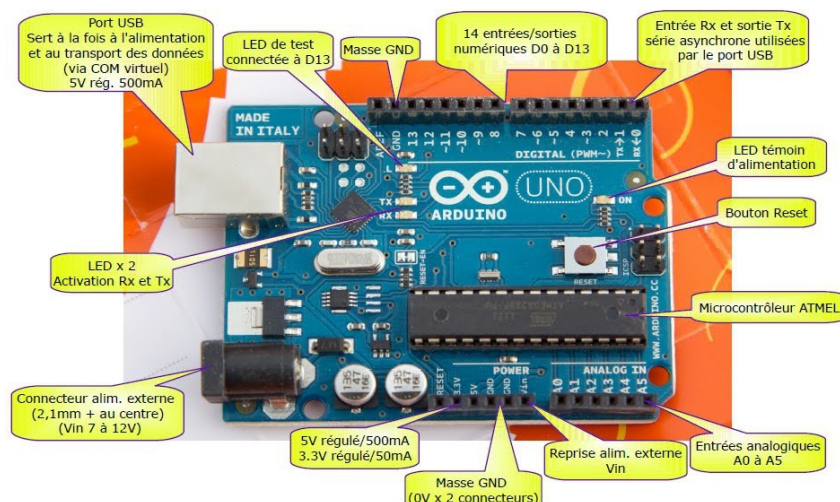


FIGURE 2.4 – Description Arduino Uno

2.6.2 Les gammes de la carte Arduino :

La famille de cartes Arduino propose une large gamme de cartes pour répondre à divers besoins et niveaux de compétence. Voici un aperçu des principales gammes :

1. Les cartes Arduino originales :

▷ **Arduino Uno :**

La carte la plus populaire et accessible, idéale pour les débutants. Elle est basée sur le microcontrôleur ATmega328 et dispose de 14 broches d'entrée/sortie numériques et 6 broches d'entrée analogique[34].



FIGURE 2.5 – Arduino Uno Board

- ▷ **Arduino Nano :** une version miniature de l'UNO, parfaite pour les projets compacts. Elle est compatible avec la plupart des Shields Uno[34].

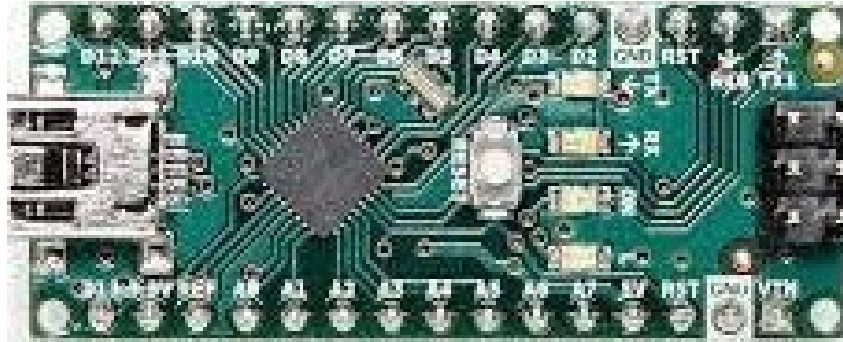


FIGURE 2.6 – Arduino Nano Board

▷ **Arduino MEGA :**

offre plus de broches d'entrée/sortie (54 numériques et 16 analogiques) que l'UNO, ce qui la rend adaptée aux projets plus complexes. Elle est basée sur le microcontrôleur ATMEGA2560.[34].



FIGURE 2.7 – Arduino MEGA Board

▷ **Arduino Due :**

Basée sur un microcontrôleur ARM Cortex-M3 plus puissant, elle offre des performances accrues pour les applications exigeantes. Elle dispose également d'un convertisseur analogique-numérique (CAN) 12 bits.[34].

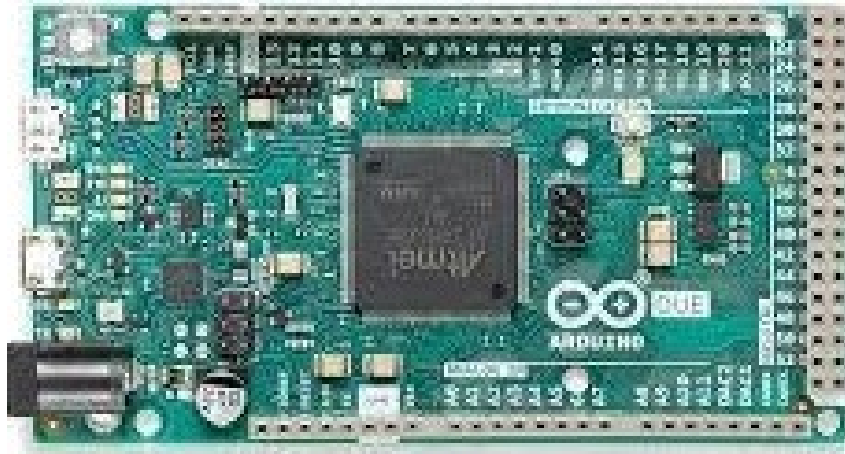


FIGURE 2.8 – Arduino Due Board

▷ **Arduino Leonardo :**

intègre une communication USB native lui permettant d’être utilisée comme souris, clavier ou joystick. Elle est idéale pour les projets d’interface homme-machine.[34].



FIGURE 2.9 – Arduino Leonardo Board

2. Les cartes Arduino MKR :

* **MKR1000 :**

Une carte Wi-Fi basée sur le microcontrôleur SAMD21. Elle est idéale pour les projets IOT (Internet des objets) qui nécessitent une connectivité sans fil.[35].



FIGURE 2.10

*** MKR Zero :**

Une carte Bluetooth basse consommation basée sur le microcontrôleur SAMD21. Elle est parfaite pour les projets portables et alimentés par batterie.[35].



FIGURE 2.11 – Arduino MKR Zero Board

*** MKR WiFi 1010 :**

Une carte Wi-Fi plus puissante que la MKR1000, basée sur le microcontrôleur ESP32. Elle offre des fonctionnalités Wi-Fi et Bluetooth.[35].



FIGURE 2.12 – Arduino MKR WiFi 1010 Board

* **MKR VIDOR 4000 :**

Une carte puissante dotée d'un FPGA (Field Programmable Gate Array) Intel Cyclone 10 CL016 et d'un microcontrôleur Cortex-M0 32 bits SAMD21. Elle est conçue pour les applications exigeantes en termes de traitement et de calcul.[35].

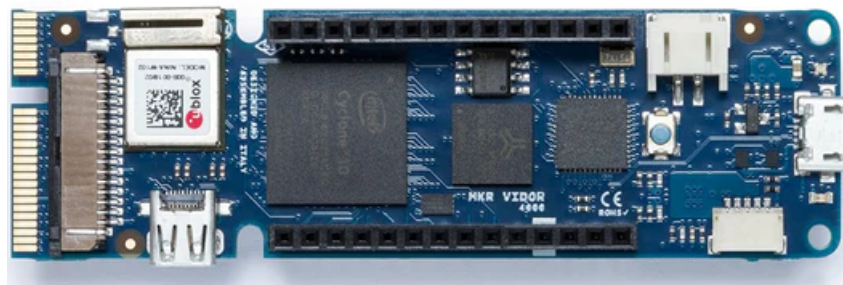


FIGURE 2.13 – Arduino MKR VIDOR 4000 BOARD

3. Les cartes Arduino Lilypad :

* **Lilypad Arduino Simple :**

Une carte flexible et imperméable conçue pour les projets portables et textiles. Elle est basée sur le microcontrôleur ATMEGA328P.[36].

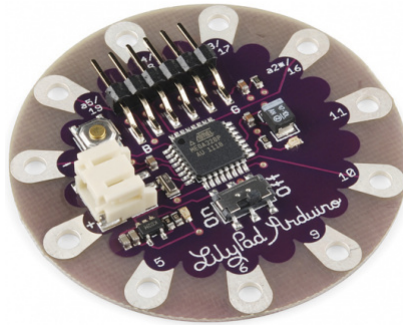


FIGURE 2.14 – Arduino Lilypad Simple Board

* **Lilypad Arduino Grande :**

Une version plus grande de la Lilypad Simple avec plus de broches et de fonctionnalités.[36].

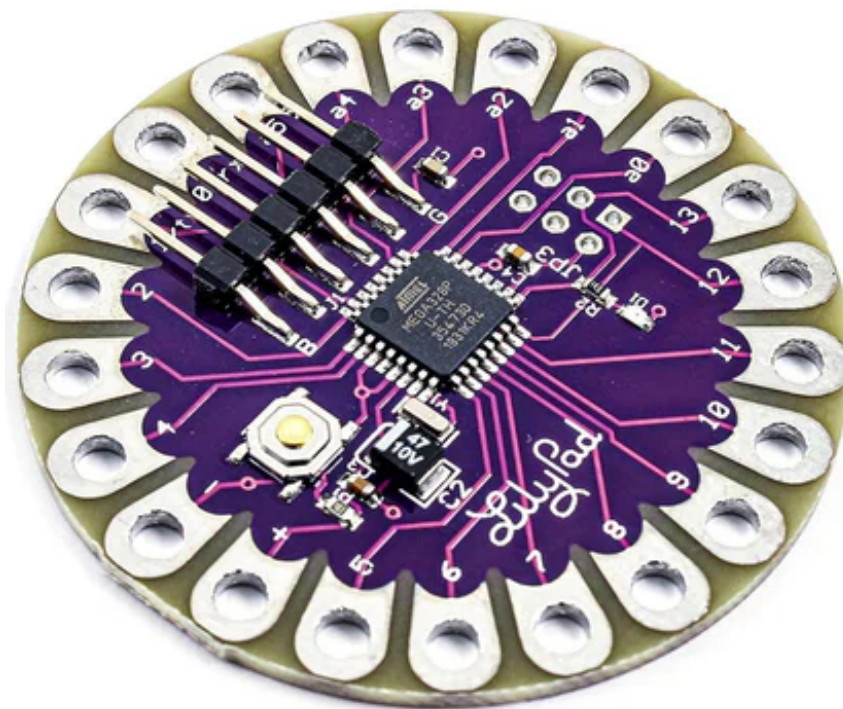


FIGURE 2.15 – Arduino Lilypad Grande Board

En plus de ces gammes principales, Arduino propose également une large gamme de cartes spécialisées pour des applications spécifiques, telles que l'impression 3D, la robotique et la domotique.

Choisir la carte Arduino adéquate dépend de vos besoins spécifiques et de votre niveau de compétence. Pour les débutants, l'Arduino Uno est un excellent choix pour sa simplicité

et sa large communauté d'utilisateurs. Pour les projets plus complexes ou nécessitant des fonctionnalités spécifiques, il existe une carte Arduino adaptée à vos besoins[33][34][35][36].

2.6.3 Les critères de choix d'Arduino Uno :

La carte Arduino UNO est un choix populaire pour la programmation électronique en raison des avantages suivants :

- **Abordabilité :**

les cartes Arduino sont relativement abordables, avec des modèles pré-assemblés coûtant moins de 2500 dinars, ce qui les rend plus économiques que d'autres plateformes[34].

- **Compatibilité multiplateforme :**

le logiciel Arduino, écrit en Java, est compatible avec Windows, Macintosh et Linux, offrant une flexibilité sur différents systèmes d'exploitation[37].

- **Environnement de programmation clair et simple :**

L'environnement de programmation Arduino (le logiciel Arduino IDE) est convivial pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.[37]

- **Logiciel et matériel Open Source :**

le logiciel Arduino et le langage Arduino sont publiés sous licence open source, disponibles pour être complétés par des programmeurs expérimentés. Le matériel, basé sur les microcontrôleurs Atmel ATMEGA8, ATMEGA168, ATMEGA328, a des schémas de modules publiés sous une licence Creative Commons, et les concepteurs de circuits expérimentés peuvent réaliser leur propre version des cartes Arduino, en les complétant et en les améliorant. Même les utilisateurs relativement inexpérimentés peuvent fabriquer la version sur plaque d'essai de la carte Arduino, dont le but est de comprendre comment elle fonctionne pour économiser le coût. [33]

TABLEAU 2.2 – Caractéristiques matérielles des cartes Arduino.

Carte	Processeur	VCC	Horloge (MHz)	AIN	DIO	PWM	USB
ArduinoBT	ATmega328	5	16	6	14	6	Aucun
Duemilanove	ATmega168	5	16	6	14	6	Normal
Duemilanove	ATmega328	5	16	6	14	6	Normal
Diecimila	ATmega168	5	16	6	14	6	Normal
Esplora	ATmega32U4	5	16	-	-	-	Micro
Ethernet	ATmega328	5	16	6	14	6	Normal
Fio	ATmega328p	3.3	8	8	14	6	Mini
Leonardo	ATmega32U4	5	16	12	20	7	Micro
LilyPad	ATmega168V	2.7- 5.5	8	6	14	6	Aucun
LilyPad	ATmega328V	2.7- 5.5	8	6	14	6	Aucun
Mega	ATmega1280	5	16	16	54	15	Normal
Mega ADK	ATmega2560	5	16	16	54	15	Normal
Mega 2560	ATmega2560	5	16	16	54	15	Normal
Micro	ATmega32U4	5	16	12	20	7	Micro
Mini	ATmega328	5	16	8	14	6	Aucun
Mini Pro	ATmega168	3.3	8	6	14	6	Aucun
Mini Pro	ATmega168	5	16	6	14	6	Aucun
<i>Suite à la page suivante</i>							

TABLEAU 2.2 – Caractéristiques matérielles des cartes Arduino[28]. (suite)

Carte	Processeur	VCC	Horloge (MHz)	AIN	DIO	PWM	USB
Nano	ATmega168	5	16	8	14	6	Mini-B
Nano	ATmega328	5	16	8	14	6	Mini-B
Pro (168)	ATmega168	3.3	8	6	14	6	Aucun
Pro (328)	ATmega328	5	16	6	14	6	Aucun
Uno	ATmega328	5	16	6	14	6	Normal
Yún	ATmega32U4	5	16	12	20	7	Hôte

2.6.4 La constitution de la carte Arduino UNO :

Un module Arduino est généralement construit autour d'un microcontrôleur ATMEL AVR et de composants complémentaires qui facilitent la programmation et l'interfaçage avec d'autres circuits. Chaque module possède au moins un régulateur linéaire 5 V et un oscillateur à quartz 16 MHz (ou un résonateur céramique dans certains modèles). Le microcontrôleur est préprogrammé avec un bootloader, de façon à ce qu'un programmeur dédié ne soit pas nécessaire.[34]

2.6.4.1 Partie matérielle :

Généralement, tout module électronique possédant une interface de programmation est basé toujours dans sa construction sur un circuit programmable ou plus.[21]

2.6.4.2 Les sources d'alimentation de la carte :

On peut distinguer deux genres de sources d'alimentation (Entrée Sortie) et cela comme suit :

- * **VIN.** La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer du 5 V de la connexion USB ou autre source 5 V régulée). On peut alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.
- * **5 V.** La tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite « tension régulée », obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte Arduino). Le 5 V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB (qui fournit du 5 V régulé) ou de toute autre source d'alimentation régulée.

- * **3V3.** Une alimentation de 3.3 V fournie par le circuit intégré FTDI (circuit intégré faisant l'adaptation du signal entre le port USB de votre ordinateur et le port série de l'ATmega) de la carte est disponible : ceci est intéressant pour certains circuits externes nécessitant cette tension au lieu du 5 V. L'intensité maximale disponible sur cette broche est de 50 mA. [35]

2.6.4.3 Les entrées & sorties :

Cette carte possède 14 broches numériques (numérotées de 0 à 13) et peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique, en utilisant les instructions `pinMode()`, `digitalWrite()` et `digitalRead()` du langage Arduino. Ces broches fonctionnent en 5 V. Chaque broche peut fournir ou recevoir un maximum de 40 mA d'intensité et dispose d'une résistance interne de « rappel au plus » (pull-up) (déconnectée par défaut) de 20-50 KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction `digitalWrite(broche, HIGH)`.

En plus, certaines broches ont des fonctions spécialisées :

- **Interruptions externes :** broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur. – Impulsion PWM (largeur d'impulsion modulée) : Broches 3, 5, 6, 9, 10, et 11. Fournissent une impulsion PWM 1 bit à l'aide de l'instruction `analogWrite()`.
- **SPI (Interface série périphérique) :** broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la librairie pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Mega.
- **I2C :** broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou interface TWI (TwoWire Interface - Interface 2 fils), disponible en utilisant la librairie `Wire/I2C` (ou TWI - Two-Wire interface - interface 2 fils).

- **LED :** Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau haut, la LED est allumée, lorsque la broche est au niveau bas, la LED est éteinte.

La carte UNO dispose de 6 entrées analogiques (numérotées de 0 à 5), chacune pouvant fournir une mesure d'une résolution de 10 bits (càd sur 1024 niveaux, soit de 0 à 1023) à l'aide de la très utile fonction `analogRead()` du langage Arduino. Par défaut, ces broches mesurent entre le 0 V (valeur 0) et le 5 V (valeur 1023), mais il est possible de modifier la référence supérieure de la plage de mesure en utilisant la broche AREF et l'instruction `analogReference()` du langage Arduino.

La carte Arduino UNO intègre un fusible qui protège le port USB de l'ordinateur contre les surcharges en intensité (le port USB est généralement limité à 500 mA en intensité). Bien que la plupart des ordinateurs aient leur propre protection interne, le fusible de la carte fournit une couche supplémentaire de protection. Si plus de 500 mA sont appliqués au port USB, le fusible de la carte coupera automatiquement la connexion jusqu'à ce que le court-circuit ou la surcharge soit stoppé.[34][35][37]

2.6.4.4 Les ports de communications :

La carte Arduino UNO a de nombreuses possibilités de communications avec l'extérieur. L'Atmega328 possède une communication série UART TTL (5 V), grâce aux broches numériques 0 (RX) et 1 (TX).

On utilise (RX) pour recevoir et (TX) pour transmettre (les données séries de niveau TTL). Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega328 programmé en convertisseur USB vers série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur. Comme un port de communication virtuel pour le logiciel sur l'ordinateur, la connexion série de l'Arduino est très pratique pour communiquer avec un PC.[34][35][37]

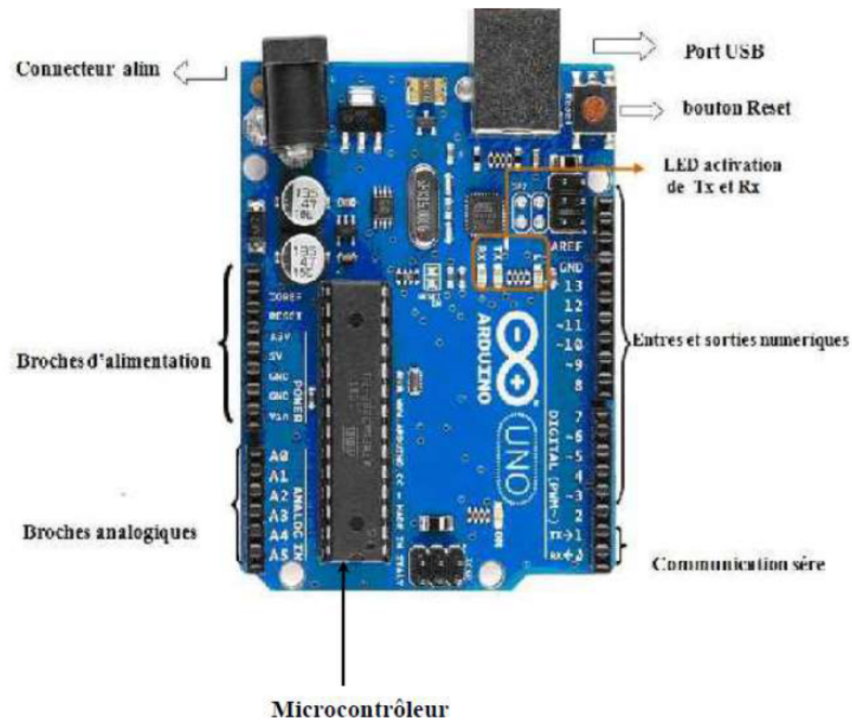


FIGURE 2.16 – Constitution de la carte Arduino UNO

2.7 Descriptions des broches de l'ATmega328P et de la carte Arduino UNO :

- **VCC (ATMEGA328P) vers 5 V (Arduino UNO) :** alimente le microcontrôleur.
- **GND (ATMEGA328P) vers GND (Arduino UNO) :** établit une référence commune de masse.
- **Port B (ATMEGA328P) vers broches numériques (Arduino UNO) :** connecte les broches d'entrée/sortie du microcontrôleur aux broches d'entrée/sortie de l'Arduino pour la communication numérique.
- **Port C (ATMEGA328P)**
 - **PC0-PC5 :** Peut être connecté aux broches numériques (Arduino UNO) pour la communication numérique.
 - **PC4 (SDA) et PC5 (SCL) :** se connectent au bus I2C pour la communication avec les périphériques I2C.

- **PC6 (ATMEGA328P) vers RESET (Arduino UNO)** : permet de réinitialiser le microcontrôleur depuis la carte Arduino
- **Port D (ATMEGA328P)**
 - **PD0 (RX) et PD1 (TX)** : Se connectent aux broches de communication série (RX et TX) sur l'Arduino UNO pour la communication série.
 - **PD2 (INT0) et PD3 (INT1)** : peuvent être utilisés comme sources d'interruption externes.
 - **PD6-PD7** : peut être utilisé pour les sorties PWM (modulation de largeur d'impulsion) [34][35][37].

Voici un schéma simplifié du brochage de l'ATmega328P sur la carte Arduino UNO :

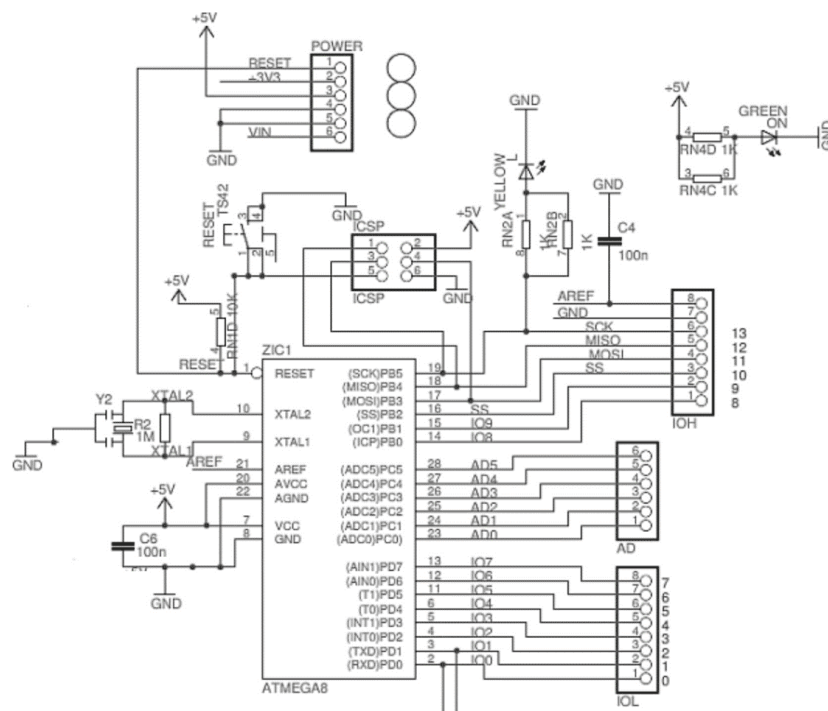


FIGURE 2.17 – Schéma simplifié de brochage ATMEGA328P avec la carte Arduino UNO

2.8 la programmation de la carte arduino UNO :

- **L'Arduino IDE :**

L'IDE est un logiciel de programmation qui permet d'écrire, de modifier un programme et de le convertir en une série d'instructions compréhensibles pour la carte. Il programme par code, contenant une cinquantaine de commandes différentes. Le langage de l'IDE Arduino est un mélange entre le C et le C++, il possède un jeu d'instructions très riche. À l'ouverture, l'interface du logiciel Arduino se présente de la façon suivante [34][35][37] :



FIGURE 2.18 – L'ouverture du logiciel

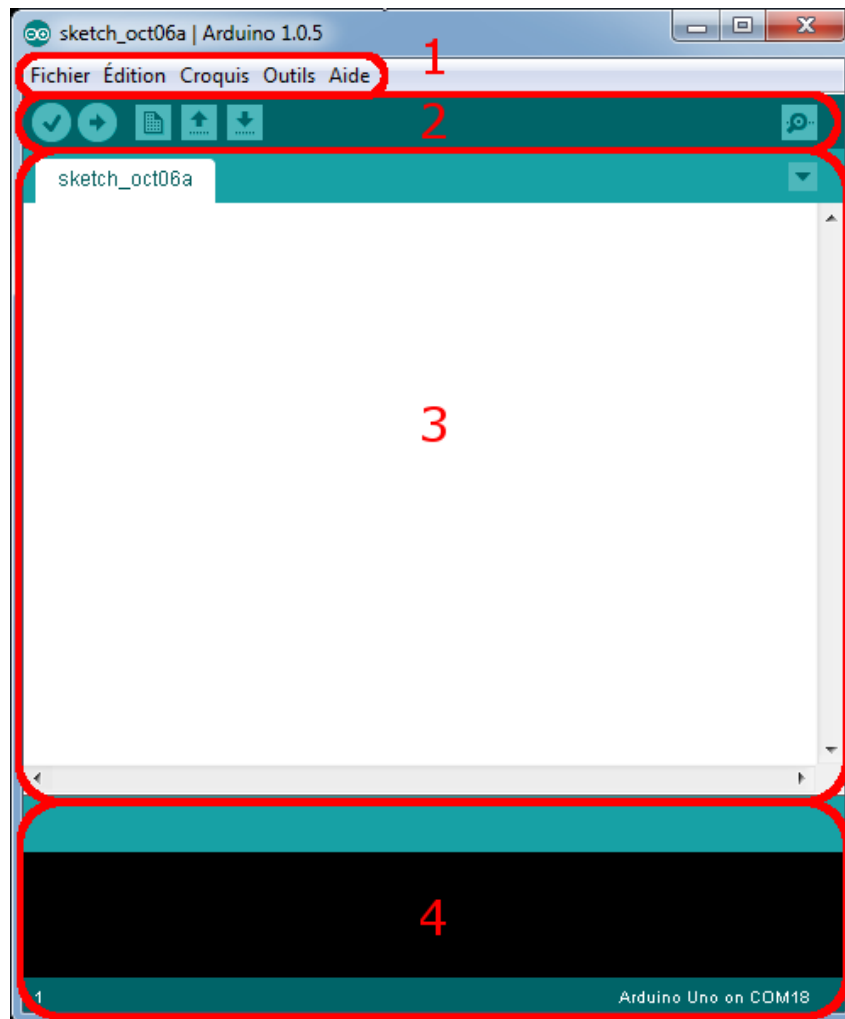


FIGURE 2.19 – La fenêtre de l'Arduino IDE.

TABLEAU 2.3 – Signification des cadres.

Cadre N°	Description
1	Ce sont les options de configuration du logiciel.
2	Il contient les boutons qui vont nous servir lorsque l'on va programmer nos cartes.
3	Ce bloc va contenir le programme que nous allons créer.
4	Celui-ci est important, car il va nous aider à corriger les fautes dans notre programme. C'est le débogueur.

- **Le menu File :**

Le menu principal sera celui que nous utiliserons le plus fréquemment. Il contient un certain nombre d'éléments qui s'avéreront très utiles. Ce menu a été progressivement traduit en français, nous allons donc examiner quelques options qui sortent de l'ordinaire.

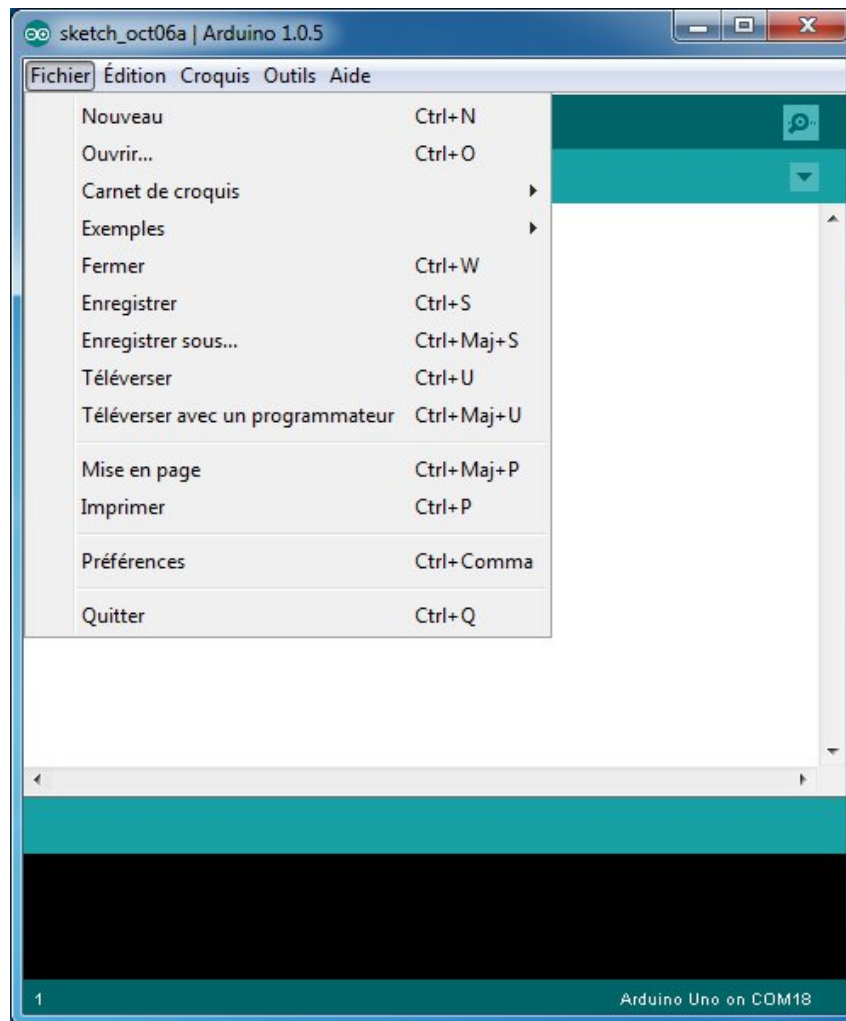


FIGURE 2.20 – Le menu Fichier

Carnet de croquis : Ce menu regroupe les fichiers que vous avez pu faire jusqu'à maintenant (et s'ils sont enregistrés dans le dossier par défaut du logiciel).

Exemples (exemples) : ceci est important, toute une liste se déroule pour afficher les noms d'exemples de programmes existants ; avec ça, vous pourrez vous aider/inspirer pour créer vos propres programmes ou tester de nouveaux composants.

Téléverser : permet d'envoyer le programme sur la carte Arduino. Nous y reviendrons.

Téléverser avec un programmeur : Idem que ci-dessus, mais avec l'utilisation d'un programmeur (vous n'en aurez que très rarement besoin).

Préférences : Vous pourrez régler ici quelques paramètres du logiciel. Le reste des menus n'est pas intéressant pour l'instant, on y reviendra plus tard, avant de commencer à programmer[34][35][37] .

- **Les boutons des outils :**

Portons maintenant notre attention sur les boutons entourés de rouge et identifiés par des étiquettes chiffrées, tels qu'illustrés par Alan sur la photo fournie.[34][35][37] .

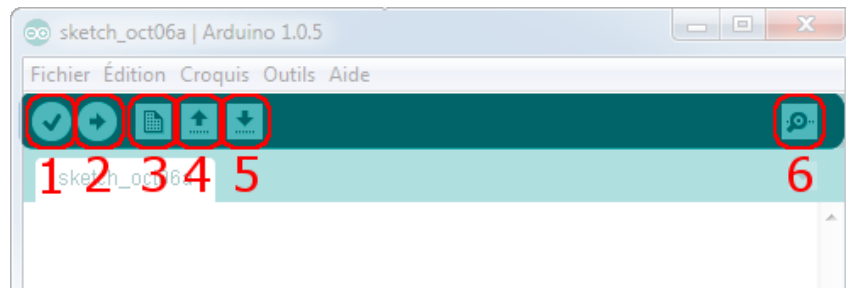


FIGURE 2.21 – La barre d'outils.

TABLEAU 2.4 – Le rôle des boutons des programmations des cartes.

Bouton	Description
1	Ce bouton permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme.
2	Charge (téléverse) le programme dans la carte Arduino.
3	Crée un nouveau fichier.
4	Ouvre un fichier.
5	Enregistre le fichier.
6	Ouvre le moniteur série (on verra plus tard ce que c'est).

- **Injection du programme** : avant d'envoyer un programme dans la carte, il est nécessaire de sélectionner le type de la carte (Arduino UNO) et le numéro de port USB (COM 4), comme à titre d'exemple cette figure 2.22 [34][35][37] .

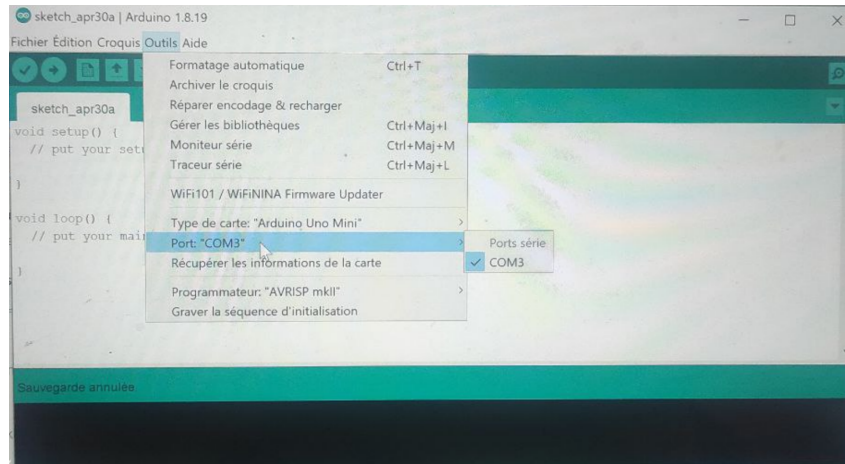


FIGURE 2.22 – Paramètre de port de carte.

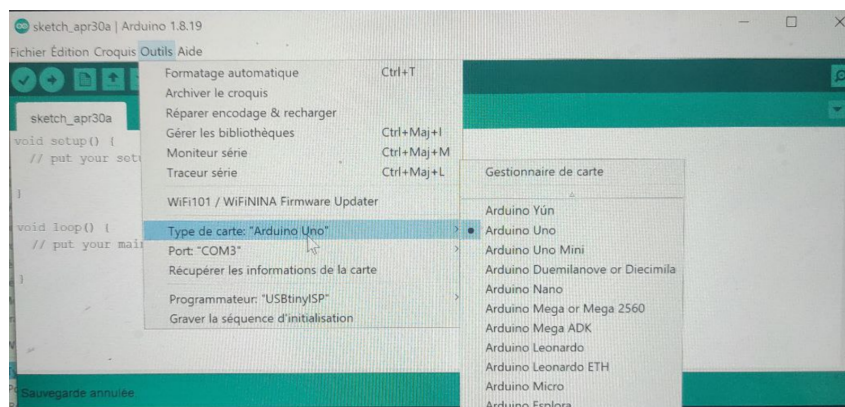


FIGURE 2.23 – Paramètre de board de la carte.

2.9 CONCLUSION :

Ce chapitre a permis d'établir une compréhension approfondie des microcontrôleurs, en particulier de l'ATmega328P qui constitue le cœur de notre plateforme ArduinoUno. Nous nous sommes lancés dans l'exploration du domaine fascinant des microcontrôleurs, en discernant leurs principales distinctions par rapport aux microprocesseurs. Nous avons mené une enquête approfondie sur l'architecture interne de l'ATMEGA328P. De plus, ce chapitre a fourni une connaissance fondamentale des outils de développement et de programmation couramment utilisés pour l'Arduino Uno, notamment des langages tels que le C et l'assembleur, ainsi que des outils de débogage et de simulation de programmes.

Chapitre

3

Composants et matériels utilisées

3.1 Introduction

Le chapitre précédent a présenté l'Arduino Uno et les microcontrôleurs. Le chapitre 3 se concentre sur les autres composants essentiels de notre système d'assistance intelligent qui fonctionneront avec l'ArduinoUno : un capteur d'empreinte digitale pour l'identification, un module de carte SD pour le stockage des données et un écran TFT pour la communication visuelle.

3.2 Empreinte digitale :

Les empreintes digitales sont le dessin formé par les lignes de la peau des doigts, appelées aussi dermatoglyphes – sont une signature que nous laissons derrière nous à chaque fois que nous touchons un objet. Les motifs dessinés par les crêtes et plis de la peau sont différents pour chaque individu ; c'est ce qui motive leur utilisation par la police criminelle depuis le 19^e siècle. On distingue deux types d'empreintes : l'empreinte directe ou visible qui laisse une marque visible à l'œil nu et l'empreinte latente ou invisible qui est composée de lipides, de sueur et de saletés déposés sur un objet touché. Une empreinte digitale se compose principalement de crêtes (ridges) et de vallées (valleys). C'est l'étude des minuties qui permet d'identifier de façon certaine un individu. [38]

3.2.1 Les caractéristiques de l'empreinte digitale :

Une empreinte digitale est une marque laissée par les crêtes des doigts, des mains, des orteils ou des pieds lorsqu'elles touchent un objet.

Il en existe deux types :

- * L'empreinte directe (qui laisse une marque visible)
- * L'empreinte latente (saleté, sueur ou autre résidu déposé sur un objet).

Les empreintes digitales sont regroupées en trois catégories principales regroupant à elles seules 95 % des doigts humains : l'arche ou l'arc (arch), le tourbillon (whorl) et la boucle

(loop). À l'intérieur de chacune de ces catégories, il y a un très grand nombre d'éléments qui nous différencient les uns des autres. En plus des cicatrices, il y a les fourches, les îlots et les espaces qui donnent un caractère unique aux empreintes latentes.

Les « boucles » constituent les motifs les plus répandus qui représentent 60 % des doigts humains : dans ce type d'empreinte, elles se replient sur elles-mêmes soit vers la droite, soit vers la gauche. Viennent ensuite les « tourbillons », qui correspondent à 30 % des doigts humains : cette empreinte, dite en verticille, comprend des lignes qui viennent s'enrouler autour d'un point, formant un genre de tourbillon. Pour finir, les motifs les moins répandus sont « les arches », qui regroupent seulement 5 % des doigts humains : cette empreinte, en arc, contient des lignes disposées les unes au-dessus des autres qui forment des A. [39]

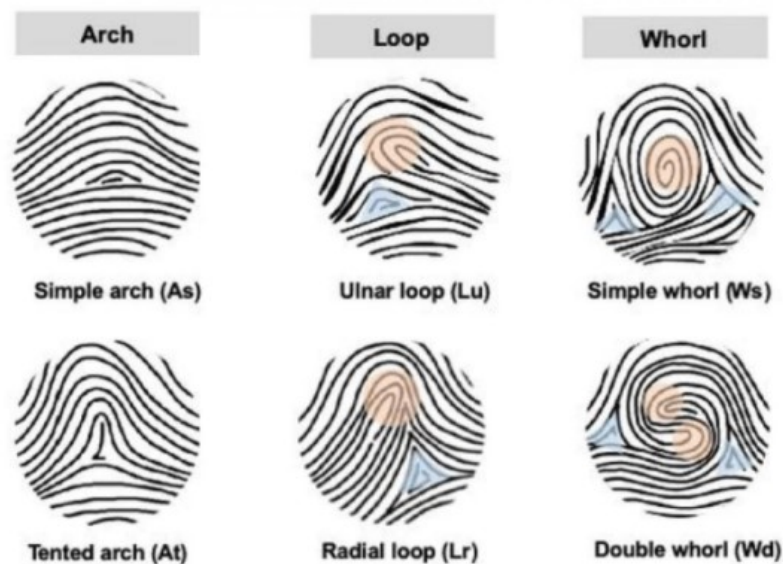


FIGURE 3.1 – Les catégories principales des empreintes digitales [3]

3.2.2 Capteurs d'empreintes digitales : Un aperçu complet

Les capteurs d'empreintes digitales sont devenus omniprésents dans le monde d'aujourd'hui, jouant un rôle crucial dans la protection de la sécurité des données et l'authentification des utilisateurs. Leurs applications s'étendent à divers domaines, notamment la sécurité publique (contrôle d'accès, enquêtes criminelles), la sécurité privée (protection des biens, accès au domicile) et les appareils personnels. Ces systèmes de reconnaissance

des empreintes digitales trouvent une large utilisation dans la sécurisation des ordinateurs, l'identification des individus dans le cadre d'enquêtes criminelles, et dans une multitude d'autres contextes.[40] [41]

3.2.3 Exploration de la technologie des capteurs :

Le marché propose une gamme captivante de capteurs d'empreintes digitales, chacun utilisant des principes distincts pour capturer les crêtes et les sillons complexes de votre empreinte digitale :

- **Capteurs optiques** : pionniers de la capture d'empreintes digitales, ils utilisent une source lumineuse pour éclairer l'empreinte et capturer une photographie numérique. Ils transforment ensuite cette image en un code binaire, créant un marqueur d'identification personnelle grâce au jeu d'ombre et de lumière à la surface de l'empreinte digitale. En raison de leur maturité et de leur rentabilité, les capteurs optiques restent un choix répandu.[39]
- **Capteurs capacitifs** : ces capteurs ingénieux exploitent la conductivité humaine pour dresser un portrait détaillé de votre empreinte digitale. Ils créent un champ électrostatique et suivent méticuleusement les caractéristiques de l'empreinte digitale en se basant sur des plaques conductrices. Les propriétés électriques des crêtes et des sillons influencent la charge stockée dans le condensateur, permettant au capteur de discerner ces variations infimes.[39]
- **Capteurs d'imagerie multispectrale** : en triomphant des conditions les plus difficiles, ces capteurs de pointe collectent des données d'empreintes digitales à la fois en surface et en subsurface. Ils réalisent cette prouesse en exploitant la puissance de plusieurs couleurs de lumière émises par des LED. Cette technologie avancée est à la base de la haute précision et de la sécurité des lecteurs Lumidigme MSI de HID.[41][42]
- **Capteurs à transistors en couches minces (TFT)** : Ces capteurs ultra-fins et à faible consommation d'énergie sont parfaitement adaptés aux appareils mobiles. Ils excellent dans la capture d'images d'empreintes digitales de haute qualité, quelles que soient les

variations d'environnement et d'éclairage, ce qui en fait une solution idéale pour les applications d'identification mobile en situation réelle.[43]

3.2.4 Structure d'un système couplé de reconnaissance d'empreintes digitales :

Un système complet et automatique de reconnaissance d'empreintes digitales est une chaîne de processus interconnectés qui, à partir de l'empreinte digitale d'un utilisateur en entrée, renvoie un résultat en sortie. Ce résultat détermine l'accès ou non de l'utilisateur à des éléments nécessitant une protection. La conception de tels systèmes a fait l'objet de nombreuses recherches et des méthodes de traitement très variées ont été proposées. Cependant, ces systèmes partagent toujours une structure commune sous-jacente. [39][40]

3.2.5 Étapes clés du processus de reconnaissance d'empreintes digitales :

Les principales étapes en images



FIGURE 3.2 – Les principales étapes en images

1. Acquisition d'image (Enrôlement) :

- L'étape initiale consiste à capturer une image numérique de l'empreinte digitale de l'utilisateur à l'aide de diverses technologies de capteurs telles que les capteurs optiques, capacitifs ou à ultrasons.[39][42]

2. Prétraitement :

L'image capturée est soumise à un prétraitement afin d'améliorer sa qualité et d'extraire les caractéristiques essentielles de l'empreinte digitale. Cela peut inclure :

- Le filtrage du bruit pour éliminer les artefacts indésirables.
- La normalisation du contraste pour améliorer la clarté.
- La segmentation des crêtes et des sillons, les motifs uniques qui forment l'empreinte digitale.[39][42]

3. Extraction de caractéristiques :

Des caractéristiques distinctives de l'empreinte digitale, appelées minutiae, sont extraites de l'image prétraitée. Les minutiae comprennent les terminaisons de crête et les bifurcations, qui constituent la base de la comparaison et de l'identification des empreintes digitales.[39][42]

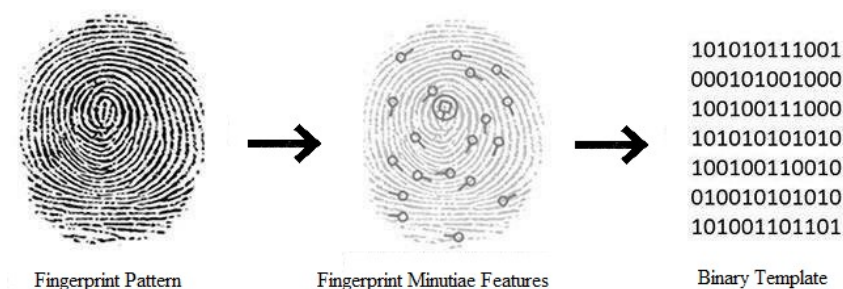


FIGURE 3.3 – Un modèle typique d'un modèle d'empreinte digitale

4. Comparaison et identification (Authentication) :

- Les minutiae extraites sont comparées à une base de données de modèles d'empreintes digitales stockés. Un modèle est une représentation numérique d'une empreinte digitale contenant un ensemble de points minutieux fiables.
- Dans un système d'identification, l'empreinte digitale extraite est comparée à tous les modèles de la base de données jusqu'à ce qu'une correspondance soit trouvée, établissant ainsi l'identité de l'utilisateur.
- Dans un système de vérification, l'empreinte digitale extraite est comparée uniquement à un modèle de référence enregistré, confirmant ou rejetant l'identité revendiquée.[39][42]

5. Décision et réponse :

Le système émet une décision finale basée sur le résultat de la comparaison :

- Si une correspondance positive est établie, l'utilisateur est autorisé à accéder aux ressources protégées ou son identité est confirmée.
- Si aucune correspondance n'est trouvée, l'accès est refusé ou l'identité de l'utilisateur est rejetée.[44]

3.2.6 Pourquoi est-ce que l'extraction basée sur des minuties dans la reconnaissance d'empreintes digitales ?

L'étape la plus importante dans la correspondance d'empreintes digitales automatique consiste à extraire de manière fiable les minutiae des images d'empreintes digitales capturées. Il existe diverses techniques d'extraction des minutiae d'empreintes digitales, classées globalement en deux types : les techniques qui fonctionnent sur des images binarisées et celles qui fonctionnent sur des images en niveaux de gris. [39][45]

3.2.7 Empreintes digitales et minutiae

Une empreinte digitale est un motif distinct de crêtes et de sillons présent sur la surface du doigt d'un individu. Une crête est définie comme un segment unique et incurvé, tandis qu'un sillon est la zone située entre deux crêtes adjacentes. Ainsi, les zones foncées de l'empreinte digitale sont appelées crêtes, et les zones blanches qui se trouvent entre elles sont appelées sillons.[39][45]

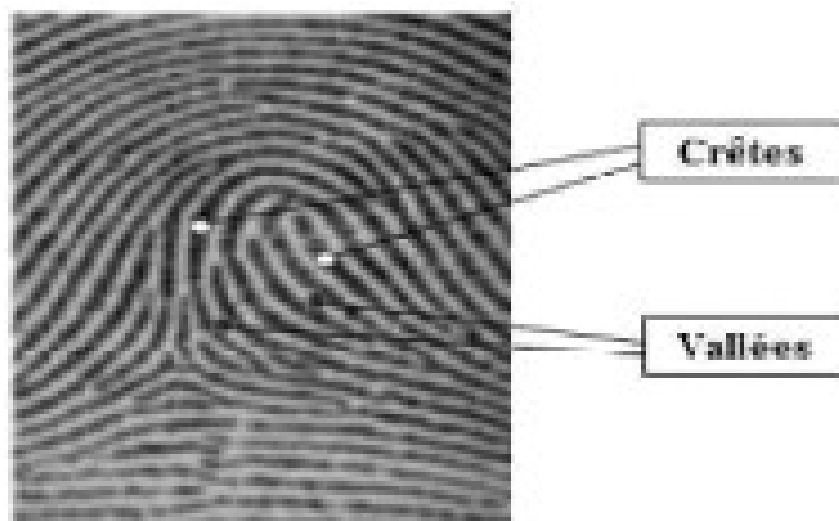


FIGURE 3.4 – Les minuties

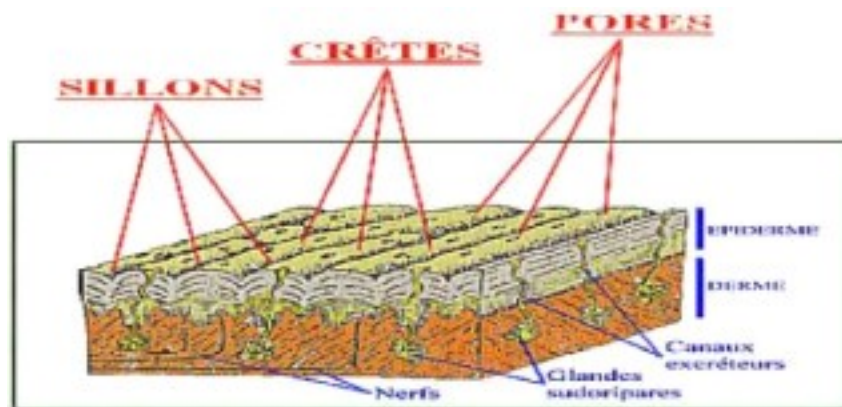


FIGURE 3.5 – Reconnaissance des empreintes digitales.

3.3 Capteur d'empreinte FPM10A :

Le capteur d'empreintes FPM10A est un capteur optique conçu pour capturer et traiter les images d'empreintes digitales pour une utilisation dans diverses applications, telles que les systèmes de sécurité, le suivi du temps et la gestion des présences, et le contrôle d'accès. Il dispose d'une sortie TTL (Transistor-Transistor Logic) série qui peut être connectée à un microcontrôleur ou à un autre appareil pour la transmission de données. [39][42]

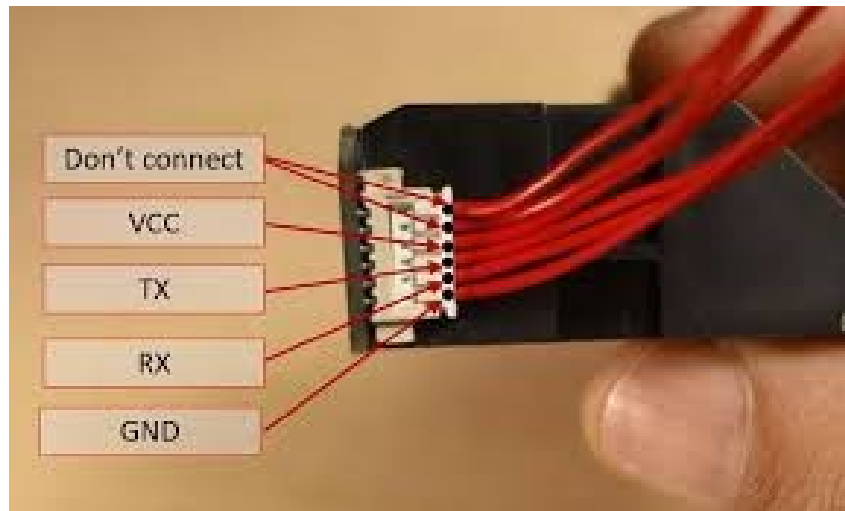


FIGURE 3.6 – Capteur d’empreinte digital.

3.3.1 Caractéristiques :

- Alimentation en tension : **DC 3,6 à 6,0 V**
- Alimentation en courant : **< 120 mA**
- Courant de crête : **< 140 mA**
- Temps d’image d’empreinte digitale : **<1.0 seconde**
- Taille de la fenêtre : **14 * 18 mm**
- Mode de correspondance : **mode de correspondance (1 :1)**
- Mode de recherche(**1 : N**)
- Fichier de signature : **256 octets**
- Fichiers de modèle : **512 octets**
- Couleur de rétroéclairage : **vert**
- Interface : **UART**
- Vitesse de communication : **9600 bps**
- Niveau de sécurité : **cinq (de bas en haut : 1,2,3,4,5)**
- Taux de faux positifs (FAR) : **<0,001%(niveau de sécurité 3)**
- Taux de faux négatifs (FRR) : **< 1, 0% (niveau de sécurité 3)**
- Capable de stocker**200 empreintes digitales différentes**

- Temps de recherche : **<1.0 seconde (1 :500, la moyenne)**
- Interface PC : **UART (niveau logique TTL) ou USB2.0 / USB1.1**
- Débit en bauds de communication (UART) : **(9600 * N) bps ou $N = 1 \sim 12$ (valeur par défaut N = 6, c'est-à-dire 57600 bps)**

Ce capteur est composé de six broches :

- * **GND : masse.**
- * **RX : Récepteur.**
- * **TX : Transmetteur.**
- * **VCC : Alimentation.**
- * **T-OUT : pin non connecté.**
- * **DNC : Pin non connecté.**

3.4 Écrans TFT : Introduction et Présentation

3.4.1 Introduction aux écrans TFT : Un aperçu

Les écrans TFT (Thin Film Transistor) sont une forme avancée de technologie d'affichage à cristaux liquides offrant une qualité d'image supérieure, des angles de vision plus larges, une consommation d'énergie réduite et une robustesse accrue par rapport aux écrans LCD traditionnels. Ils sont largement utilisés dans les moniteurs, les écrans d'ordinateur et les appareils portables pour leur haute résolution, leur précision des couleurs et leur rapport de contraste élevé. Les écrans TFT offrent une expérience utilisateur améliorée grâce à leurs fonctionnalités tactiles et sont disponibles sur le marché en tant qu'option abordable pour les utilisateurs soucieux de leur budget, offrant une qualité d'image constante et une interaction intuitive. [46]

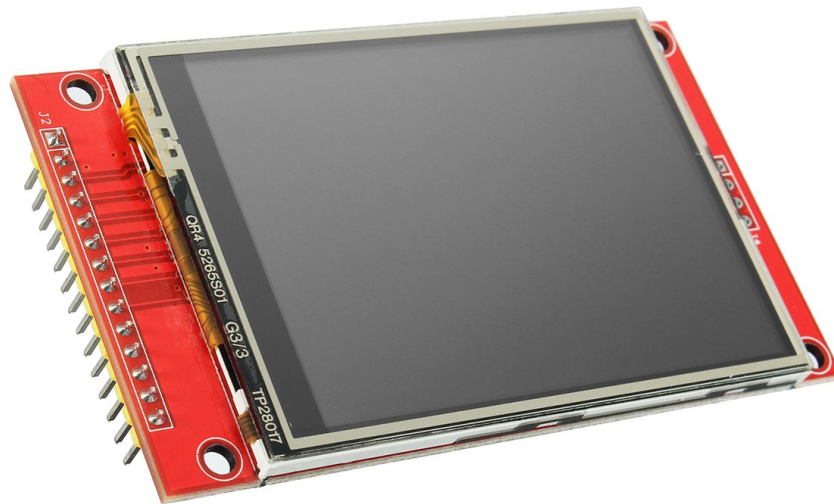


FIGURE 3.7 – ECRAN TFT

3.4.2 Structure des écrans TFT :

Les écrans TFT se composent de deux principaux composants : un substrat de verre contenant des TFT et un autre substrat de verre avec un filtre de couleur (RGB). Les TFT sont des transistors qui contrôlent activement les pixels individuellement, améliorant les performances de l'affichage. La structure de l'affichage comprend des réflecteurs pour gérer l'émission de lumière, assurant une illumination uniforme sur l'ensemble de l'écran. Les couches de cellules indépendantes recouvertes de filtres de couleur sont utilisées, chaque ensemble de couleurs étant affecté à un transistor dédié.[47]

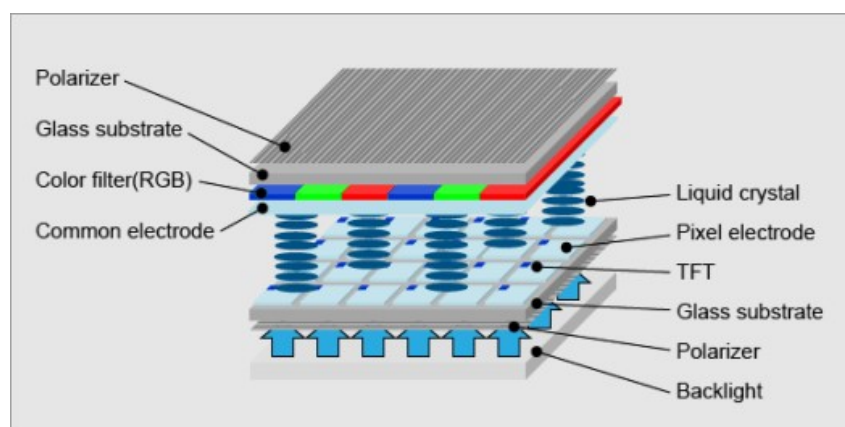


FIGURE 3.8 – Structure de l'écran LCD TFT TN

3.4.3 Fonctionnement des écrans TFT :

Le fonctionnement des écrans TFT implique la transmission d'informations aux cristaux qui modifient leur orientation, permettant à la lumière de passer par le filtre de couleur souhaité. Cette technologie d'affichage à matrice active assure une performance supérieure par rapport aux affichages à matrice passive.[48]

3.4.4 Types d'écrans TFT :

Les écrans TFT sont disponibles en différents types, chacun avec ses propres caractéristiques[49] :

- **TN (Nématique torsadée) + film** : Ce type d'écran TFT offre des temps de réponse rapides, le rendant adapté aux applications nécessitant des visuels à haute vitesse.
- **Super TFT (IPS)** : Les écrans IPS offrent des angles de vue et une reproduction des couleurs supérieurs, les rendant idéaux pour les applications nécessitant une représentation précise des couleurs.
- **Alignement vertical multi-domaine (MVA)** : les écrans MVA offrent des rapports de contraste et des angles de vue élevés, les rendant adaptés aux applications nécessitant des visuels de haute qualité.

3.4.5 Avantages des écrans TFT :

Avantages par rapport aux autres technologies d'affichage[50] :

- * **Conception compacte** : Les écrans TFT sont compacts et légers, les rendant adaptés aux appareils portables.
- * **Faible émission de chaleur** : Les écrans TFT émettent peu de chaleur, réduisant le besoin de systèmes de refroidissement.
- * **Faibles émissions électromagnétiques** : Les écrans TFT émettent des niveaux faibles de rayonnement électromagnétique, minimisant les risques potentiels pour la santé.

- * **Fatigue visuelle réduite** : Les écrans TFT réduisent la fatigue visuelle, les rendant adaptés à une utilisation prolongée.

3.4.6 Inconvénients des écrans TFT :

Malgré leurs avantages, les écrans TFT peuvent faire face à certains inconvénients [50] :

- **Manque de luminosité** : Les écrans TFT peuvent avoir des difficultés avec la luminosité dans certaines applications.
- **Limitation de la disponibilité des résolutions élevées** : Les écrans TFT peuvent avoir une disponibilité limitée de résolutions très élevées pour une utilisation professionnelle.

[11]

3.4.7 Présentation d'un module LCD 2,4 pouces :

Ce module d'affichage TFT de 2,4 pouces avec une résolution de 240×320 utilise l'interface SPI pour la communication. L'écran LCD dispose d'un contrôleur interne avec des fonctions de base permettant de dessiner des points, des lignes, des cercles et des rectangles et d'afficher les textes.[51]

Description du produit :

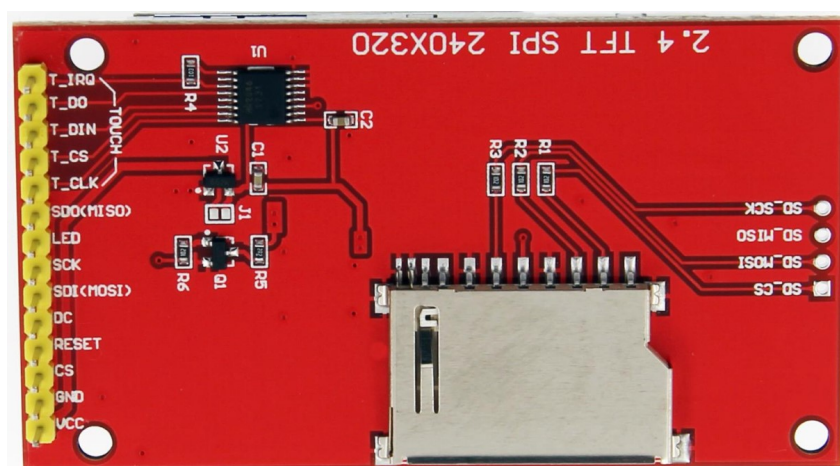


FIGURE 3.9 – Pin image d'écran.

Le module LCD utilise une méthode de communication SPI à 4 fils avec un circuit intégré de driver ILI9341, une résolution de 240×320 et une fonction tactile (en option). Le module comprend un écran LCD, un circuit de commande du rétroéclairage et un circuit de commande de l'écran tactile.

3.4.8 Fonctionnalités du produit :

1. Écran couleur 2,4 pouces, prend en charge l'affichage de 65 000 couleurs 16 bits, affiche des couleurs riches.
2. Résolution 240×320 avec fonction tactile en option
3. Utilisant le bus série SPI, il ne prend que quelques E/S pour éclairer l'écran.
4. Extension facile de l'expérience avec un emplacement pour carte SD
5. Fournit un riche programme d'exemple.
6. Normes de processus de qualité militaire, travail stable à long terme
7. Fournit un support technique pour les pilotes sous-jacents.[52]

3.5 Module de carte SD

Le module de carte SD est un composant fondamental de notre projet, offrant une solution fiable et rentable pour stocker et accéder aux données.[53]

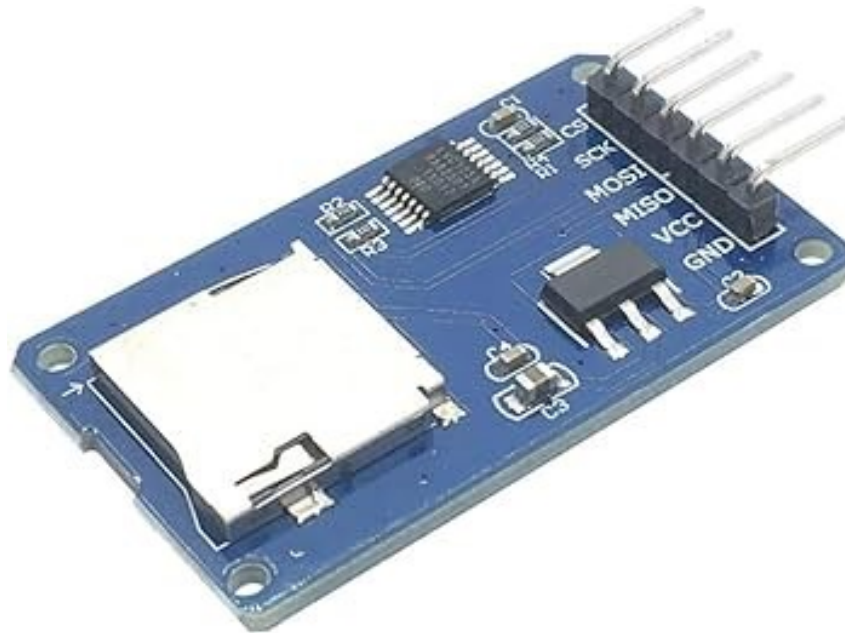


FIGURE 3.10 – Module de carte mémoire.

3.5.1 Qu'est-ce qu'un module de carte SD ?

Un module de carte SD est un circuit imprimé qui permet l'utilisation de cartes SD dans des projets électroniques, permettant la lecture et l'écriture de données avec un microcontrôleur. Il est compatible avec divers formats de cartes SD, notamment SD, SDHC et SDXC, et est essentiel pour les projets nécessitant un enregistrement ou un stockage de données.[53]

3.5.2 Composants d'un module de carte SD

Le module de carte SD est composé de plusieurs composants, notamment [53] :

- **Fente de carte SD** : Le module dispose d'une fente pour insérer des cartes SD, permettant le transfert de données entre le module et la carte.
- **Régulateur de tension** : Le module dispose d'un régulateur de tension intégré qui fournit une alimentation électrique de 3,3 V CC stable à la carte SD.
- **Broches de commande** : Le module dispose de plusieurs broches de commande, notamment GND, VCC, MOSI, MISO, SCK et CS, qui sont utilisées pour établir des connexions entre le module et le microcontrôleur.

- **Interface SPI ou SDIO :** Le module utilise soit l'interface SPI soit l'interface SDIO pour le transfert de données, selon les capacités du microcontrôleur.

3.5.3 Fonctionnement :

Les modules de carte SD fonctionnent sur une tension continue de 3,3 V, et toute tension supérieure à 3,6 V peut endommager définitivement la carte SD. Pour éviter cela, le module dispose d'un régulateur de tension de carte qui garantit le niveau de tension approprié. Le module peut être interfacé avec le microcontrôleur en mode SPI (Serial Peripheral Interface) ou SDIO (Secure Digital Input-Output). Le mode SPI est plus facile à utiliser, tandis que le mode SDIO est plus rapide et est utilisé dans les téléphones mobiles et les appareils photo numériques. Pour connecter le module au microcontrôleur, la broche GND de l'Arduino doit être connectée à la broche GND du module, la broche VCC du module doit être connectée à la broche 5 V de l'Arduino, et les broches MOSI, MISO, SCK et CS doivent être connectées aux broches correspondantes de l'Arduino. [53]

3.6 Module DS3231 – Horloge en Temps Réel RTC :

Le module DS3231 est une horloge en temps réel (RTC) de haute précision, souvent utilisée dans les projets électroniques pour fournir des fonctions de calendrier et d'horloge en temps réel.[35]



FIGURE 3.11 – Module DS3231

3.6.1 Caractéristiques et Spécifications :

Le module DS3231 est une horloge très précise qui utilise un oscillateur à quartz compensé en température (TCXO) pour fournir un signal d'horloge avec un haut degré de précision. Le module comprend un système de secours alimenté par batterie intégré qui garantit que l'horloge continue de fonctionner même lorsque l'alimentation principale est déconnectée. Le module comprend également un oscillateur à quartz de 32,768 kHz, qui fournit le signal d'horloge pour le RTC.

Le module DS3231 comprend une mémoire EEPROM intégrée de 128 octets, qui peut être utilisée pour stocker diverses données, telles que la date et l'heure actuelles, les paramètres d'alarme et d'autres données définies par l'utilisateur. Le module comprend également un circuit oscillateur intégré qui compense les changements de température, ce qui garantit une grande précision sur une large plage de températures.[35]

3.6.2 Interface avec les Microcontrôleurs :

Le module DS3231 peut être facilement interfacé avec des microcontrôleurs en utilisant le protocole de communication I2C. Le module dispose d'une adresse I2C unique de 0x68,

qui peut être utilisée pour communiquer avec le module. Le module peut être connecté au microcontrôleur à l'aide de deux fils, l'un pour les données et l'autre pour l'horloge.

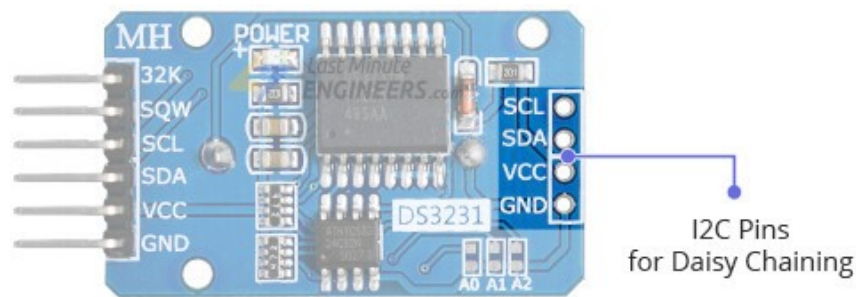


FIGURE 3.12 – Interface I2C de DS3231[15]

Le module DS3231 comprend plusieurs registres qui peuvent être accessibles à l'aide du protocole I2C. Ces registres incluent les registres d'horloge et de calendrier, les registres d'alarme et les registres de contrôle et de statut. Les registres d'horloge et de calendrier stockent la date et l'heure actuelles, tandis que les registres d'alarme peuvent être utilisés pour définir des alarmes pour des moments ou des événements spécifiques. Les registres de contrôle et de statut fournissent des informations sur l'état du module et peuvent être utilisés pour contrôler diverses fonctions du module.[35]

3.6.3 Applications :

Le module DS3231 est couramment utilisé dans divers projets électroniques, tels que les enregistreurs de données, les appareils photo à intervalle de temps et d'autres projets nécessitant un suivi précis du temps. Le module est également utilisé dans des applications industrielles, telles que les systèmes de contrôle, où une précision horaire est essentielle. [35]

3.7 conclusion :

Ce chapitre a mis en lumière les composants électroniques essentiels qui composent le système d'assistance intelligent et leur interaction avec l'Arduino Uno pour assurer un fonctionnement optimal.

Le capteur d'empreintes digitales FPM10A joue un rôle crucial dans l'identification des utilisateurs grâce à sa technologie de balayage optique. Les écrans LCD TFT, quant à eux, offrent une interface utilisateur intuitive et conviviale avec des couleurs vives, un écran tactile et une connexion simple via SPI.

Le module de carte SD comble les limitations de stockage de l'Arduino en permettant de stocker des données volumineuses telles que les empreintes digitales, les horaires et les paramètres du système.

Enfin, le module RTC DS3231 garantit un horodatage précis et fiable, indispensable pour le suivi des présences, la planification et la gestion des données temporelles.

En posant les bases solides d'un système d'assistance intelligent performant et polyvalent, ce chapitre ouvre la voie au chapitre suivant qui explorera les étapes de mise en œuvre et de prototypage. Cette phase cruciale permettra de concrétiser la conception théorique et d'évaluer les performances du système dans des conditions réelles.

Chapitre

4

Partie pratique

4.1 Introduction

Après avoir identifié les composants électroniques clés dans le chapitre 3, nous sommes maintenant prêts à passer à la prochaine étape : la mise en place et la création d'un prototype du système d'assistance intelligent. Ce chapitre examinera la planification du système en marche.

Ce chapitre aura pour but de développer un prototype afin d'évaluer les performances du système et de détecter d'éventuels problèmes. Nous allons évaluer les diverses fonctionnalités de l'assistant intelligent, en nous assurant qu'il satisfait aux exigences établies.

4.2 Description du système

Le prototype réalisé se compose de six blocs principaux : l'alimentation (tension régulée), la détection biométrique (lecteur d'empreintes FPM10A), l'affichage (écran TFT ST7789), le stockage de données (module de carte SD), la gestion du temps (horloge RTC DS3231) et la gestion du système (microcontrôleur ATMEGA 328P) (voir figure 4.1). Ce système est conçu pour fournir des informations médicales cruciales aux professionnels de santé en cas d'urgence, améliorant ainsi les délais d'intervention et contribuant à sauver des vies. Il stocke des informations essentielles telles que le groupe sanguin, l'âge, les maladies chroniques et les médicaments en cours, et comprend des interfaces conviviales, des mesures de sécurité et des fonctionnalités telles que l'authentification biométrique et les contacts d'urgence.

Dans ce système, il est toujours supposé que le patient est inconscient. Par conséquent, c'est le professionnel de santé qui scanne les empreintes digitales du patient pour accéder à ses informations médicales. Lorsqu'une empreinte digitale est scannée, les informations médicales du patient sont affichées sur l'écran TFT, permettant un accès rapide et efficace aux données critiques.

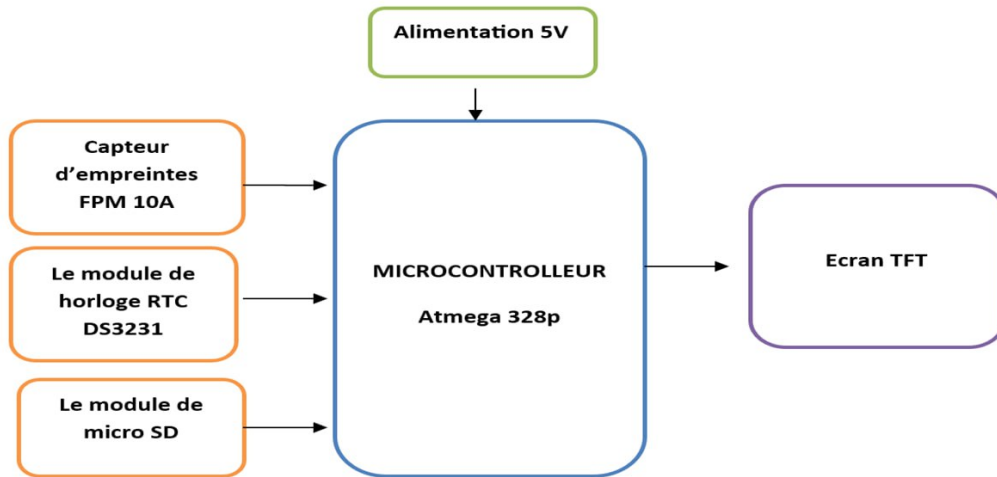


FIGURE 4.1 – Schéma bloc du système

La figure suivante montre des photos du système.



FIGURE 4.2 – L'extérieur de notre appareil

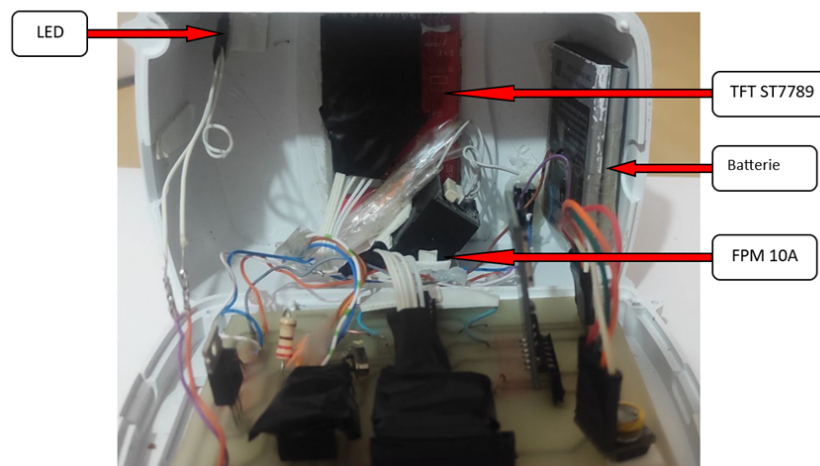


FIGURE 4.3 – L'extérieur du boîtier de notre appareil

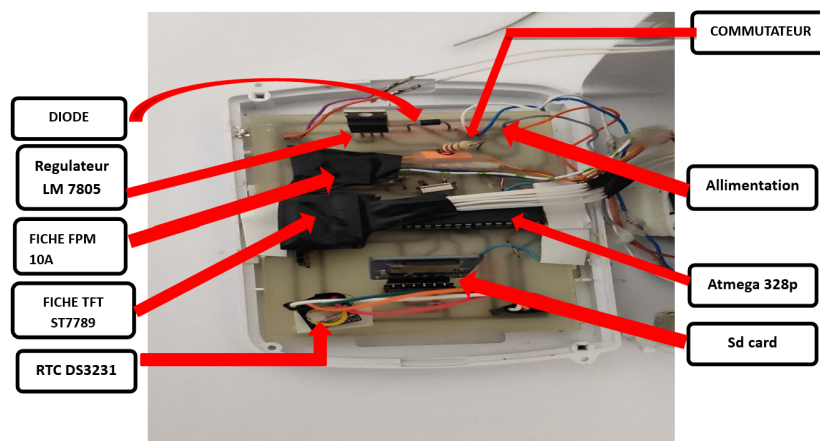


FIGURE 4.4 – L'intérieur du boîtier de notre appareil

4.3 Schéma électrique du montage

La figure 4.5 présente le schéma électrique de notre prototype, conçu à l'aide du logiciel Proteus. Ce schéma illustre clairement les différents composants qui constituent le circuit.

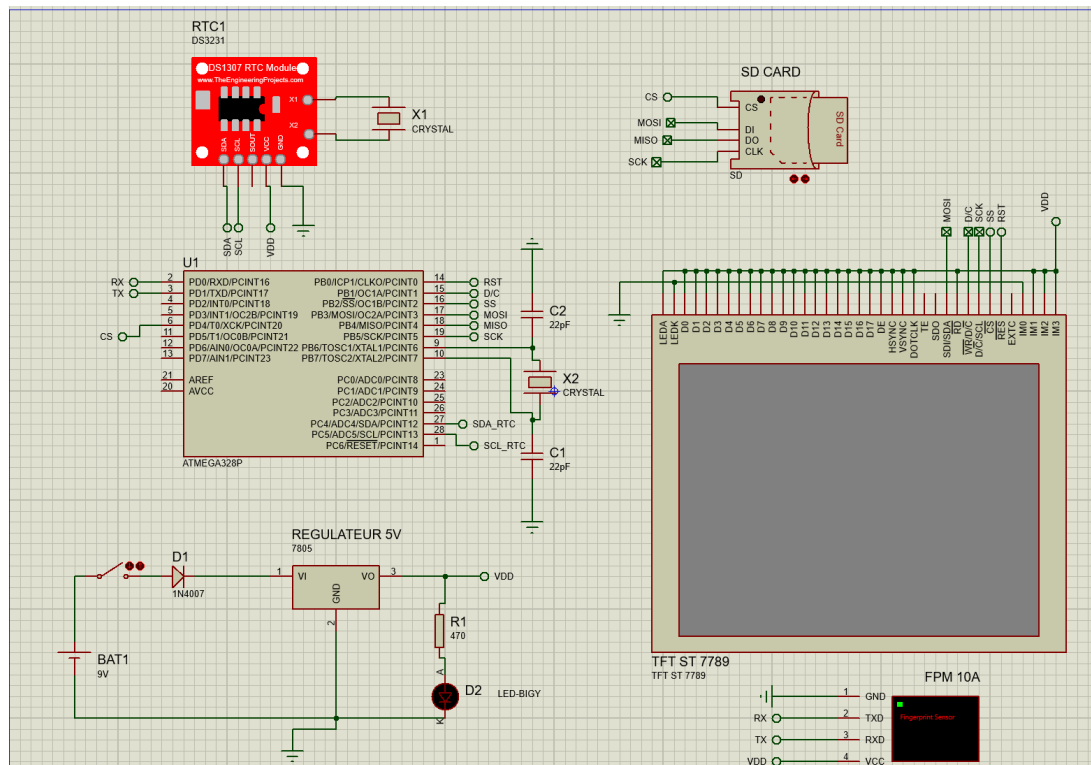


FIGURE 4.5 – Schéma électrique final sous Proteus

Les connexions électriques entre ces composants sont réalisées à l'aide de fils et de pistes sur le PCB, avec Proteus facilitant la création de ces connexions en permettant une disposition claire et organisée. Avant la fabrication du prototype physique, le schéma est simulé dans Proteus pour vérifier le bon fonctionnement du circuit, permettant ainsi de détecter et de corriger d'éventuelles erreurs de conception.

4.4 Fonctionnements du système

Le fonctionnement du système d'assistance intelligent repose sur l'interaction coordonnée de ses différents composants. Voici une description détaillée de chaque étape du processus :

1. Alimentation (Tension régulée)

Le système est alimenté par une source de tension régulée, assurant une alimentation stable et fiable pour tous les composants électroniques. Cette alimentation est essentielle pour le bon fonctionnement du lecteur d'empreintes, de l'écran TFT, du module de carte SD, de l'horloge RTC et du microcontrôleur.

2. Détection biométrique (lecteur d'empreintes FPM10A)

Lorsqu'un professionnel de santé scanne l'empreinte digitale du patient à l'aide du lecteur d'empreintes FPM10A, le microcontrôleur ATMEGA 328P traite les données biométriques pour vérifier l'identité du patient.. Le lecteur d'empreintes utilise des modèles mathématiques des empreintes digitales, ce qui empêche la reconstitution des images originales, assurant ainsi la sécurité des données biométriques.

3. Affichage des données (Écran TFT ST7789)

Une fois l'empreinte digitale validée, le microcontrôleur récupère les informations médicales stockées sur le module de carte SD. Ces informations sont ensuite affichées sur l'écran TFT ST7789, permettant au professionnel de santé de consulter rapidement les données critiques telles que le groupe sanguin, les maladies chroniques et les médicaments en cours. L'écran TFT offre une bonne visibilité et une interface conviviale pour une utilisation efficace en situation d'urgence.

4. Gestion du Temps (horloge RTC DS3231)

Le module RTC DS3231 assure une chronologie précise et fiable, même en cas de coupure de courant. Une batterie de secours intégrée protège l'horloge interne du système, garantissant le maintien du temps.

Lors de la vérification réussie de l'empreinte digitale du patient, la date et l'heure actuelles s'affichent sur l'écran TFT.

5. Stockage des données (Module de Carte SD)

Le module de carte SD stocke toutes les informations médicales du patient de manière sécurisée. Le microcontrôleur gère les opérations de lecture et d'écriture sur la carte SD, garantissant que les données sont à jour et accessibles en cas de besoin. La sécurité des données est renforcée par des mécanismes de contrôle d'accès et de chiffrement pour protéger contre les accès non autorisés et les pertes de données.

4.5 Étude détaillée du système

4.5.1 Lecteur d'empreintes :

L'élément principal de détection utilisé dans notre prototype est le capteur d'empreintes digitales FPM10A. Ce capteur est relié au microcontrôleur ATmega328P via une interface de communication UART/TTL. Lorsqu'un professionnel de santé scanne l'empreinte digitale du patient, le capteur capture l'image de l'empreinte et la transmet au microcontrôleur pour traitement et vérification.

Connexions :

- VCC du capteur FPM10A est relié à la source d'alimentation de 3,3 V à 6 V.
- GND du capteur FPM10A est relié au GND du microcontrôleur.
- TX du capteur FPM10A est relié à la broche RX (Digital Pin 2) du microcontrôleur.
- RX du capteur FPM10A est relié à la broche TX (Digital Pin 3) du microcontrôleur.

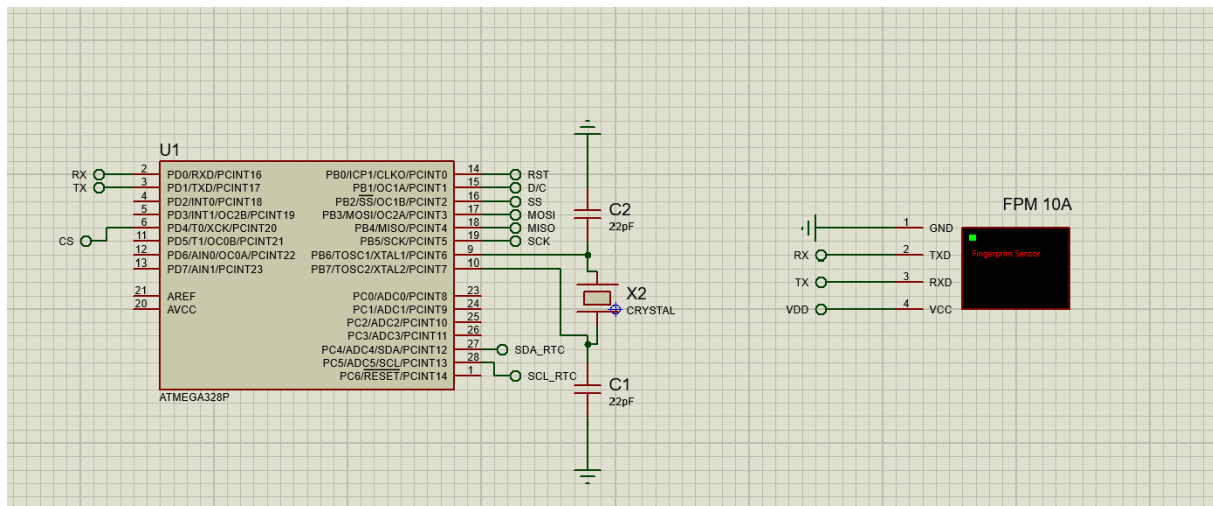


FIGURE 4.6 – Schéma du module FPM10A sous Proteus

4.5.1.1 Partie du programme lecteur d'empreintes :

Pour programmer efficacement le capteur d'empreintes digitales FPM10A, il est essentiel de suivre une série d'étapes méthodiques, comme indiqué dans l'annexe. Ces étapes comprennent la définition des broches pour la communication série, telles que les broches 2 et 3 pour RX et TX, l'utilisation de la bibliothèque Adafruit Fingerprint Sensor pour créer une instance représentant le capteur, et l'initialisation du capteur pour qu'il soit prêt à lire et traiter les empreintes digitales. Dans le cadre de notre système, nous avons également mis en place la fonction `getFingerprintIDez()` pour identifier de manière fiable et rapide les utilisateurs en extrayant l'identifiant unique de leurs empreintes digitales à l'aide d'un capteur spécifique. Le processus opérationnel de cette fonction est détaillé comme suit :

- * **Capture de l'image :** La procédure commence par la capture d'une image de l'empreinte digitale via `finger.getImage()`. Si cette opération échoue, indiquée par un retour différent de `FINGERPRINT_OK`, la fonction s'interrompt immédiatement et renvoie -1, signalant une erreur dans la capture.
- * **Conversion en modèle :** L'étape suivante consiste à transformer l'image capturée en un modèle numérique à l'aide de `finger.image2Tz()`. Un échec à cette étape entraîne également un retour de -1, indiquant un problème dans la conversion de l'image en modèle.
- * **Recherche dans la base de données :** Le modèle est ensuite rapidement comparé aux empreintes stockées dans la base de données grâce à `finger.fingerFastSearch()`. Si aucune correspondance n'est trouvée, comme indiqué par un retour différent de `FINGERPRINT_OK`, un message encourageant à réessayer est affiché et -1 est renvoyé.
- * **Identification réussie :** lorsqu'une correspondance est identifiée, un message de confirmation est affiché, l'identifiant de l'empreinte digitale est montré sur le moniteur série, et après une courte attente de 3 secondes, cet identifiant est retourné par la fonction.

Ce processus illustre l'approche systématique et efficace adoptée pour l'intégration et l'utilisation du capteur d'empreintes digitales FPM10A dans notre système, garantissant une identification précise et rapide des utilisateurs.

4.5.2 L'affichage TFT ST7789

L'affichage TFT ST7789 est un composant clé de notre système d'assistance intelligent, permettant de visualiser les informations médicales du patient de manière claire et efficace. Ce module d'affichage utilise le protocole de communication SPI et offre une résolution de 240x240 pixels, ce qui permet d'afficher des images et des textes avec une grande netteté.

Pour connecter l'affichage TFT ST7789 à un microcontrôleur ATmega328P (par exemple, sur une carte Arduino Uno), suivez les connexions suivantes :

- **VCC** de l'afficher TFT relié à la source d'alimentation de 3,3 V à 6 V.
- **GND** de l'affichage TFT relié au GND du microcontrôleur.
- **Sck** de l'affichage TFT relié à la broche SCK (Digital Pin 19) du microcontrôleur.
- **RES** de l'affichage TFT relié à la broche RES (Digital Pin 14) du microcontrôleur.
- **SS** de l'affichage TFT relié à la broche SS (Digital Pin 16) du microcontrôleur.
- **MOSI** de l'affichage TFT relié à la broche MOSI (Digital Pin 17) du microcontrôleur.
- **DC** de l'affichage TFT relié à la broche DC (Digital Pin 9) du microcontrôleur.

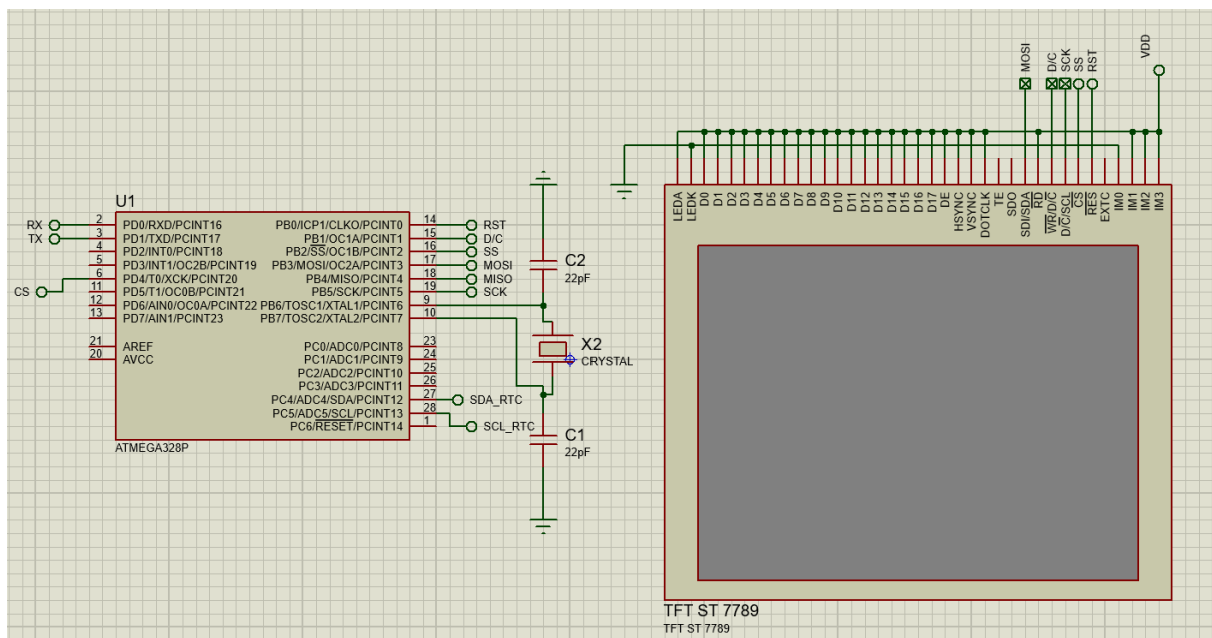


FIGURE 4.7 – Schéma de l'écran LCD TFT sous Proteus

4.5.2.1 Programme d'affichage :

La fonction `displayFileContent(int fingerprintID)` est spécialement conçue pour afficher le contenu d'un fichier texte sur un écran TFT, en se basant sur l'identifiant unique d'une empreinte digitale. Ce processus débute par la génération du nom du fichier, qui est créé en concaténant "p", suivi de l'identifiant de l'empreinte digitale converti en chaîne de caractères, et enfin, de l'extension ".txt." Une fois le nom du fichier établi, le fichier correspondant est recherché et ouvert depuis la carte SD. Si le fichier est localisé et ouvert avec succès, l'écran TFT est préalablement nettoyé en utilisant la commande `tft.fillScreen(ST77XX_BLACK)`, afin d'établir un fond noir. Par la suite, les paramètres du texte, tels que la position du curseur, la couleur et la taille du texte, sont définis. Le contenu du fichier est alors lu et affiché caractère par caractère. Lorsqu'un caractère de nouvelle ligne (`\n`) est rencontré, le curseur est déplacé à la ligne suivante pour garantir un affichage correct du texte. Si le caractère lu n'est pas une nouvelle ligne, il est directement affiché à l'écran. Une fois la lecture et l'affichage du contenu terminés, le fichier est fermé. Dans le cas où le fichier spécifié n'est pas trouvé, un message d'erreur " Personne introuvable " est affiché, indiquant ainsi que l'identifiant d'empreinte digitale fourni ne correspond à aucun fichier sur la carte SD. Cette fonctionnalité offre une interaction utilisateur personnalisée et enrichit l'expérience utilisateur en la rendant à la fois interactive et sécurisée, grâce à l'utilisation de l'identification biométrique.

4.5.3 Module de SD Card

Le module de carte SD est un composant essentiel de notre système d'assistance intelligent, permettant de stocker et de récupérer des informations médicales cruciales. Ce module utilise le protocole de communication SPI pour interagir avec le microcontrôleur ATmega328P.

Pour connecter le module de carte SD à un microcontrôleur ATmega328P (par exemple, sur une carte Arduino Uno), suivez les connexions suivantes :

- * **CS** de la carte SD relié à la broche CS (Digital Pin 6) du microcontrôleur.
- * **MOSI** de la carte SD relié à la broche MOSI (Digital Pin17) du microcontrôleur.
- * **MISO** de la carte SD relié à la broche MISO (Digital Pin 18) du microcontrôleur.

* **Sck** de la carte SD relié à la broche SCK (Digital Pin 19) du microcontrôleur.

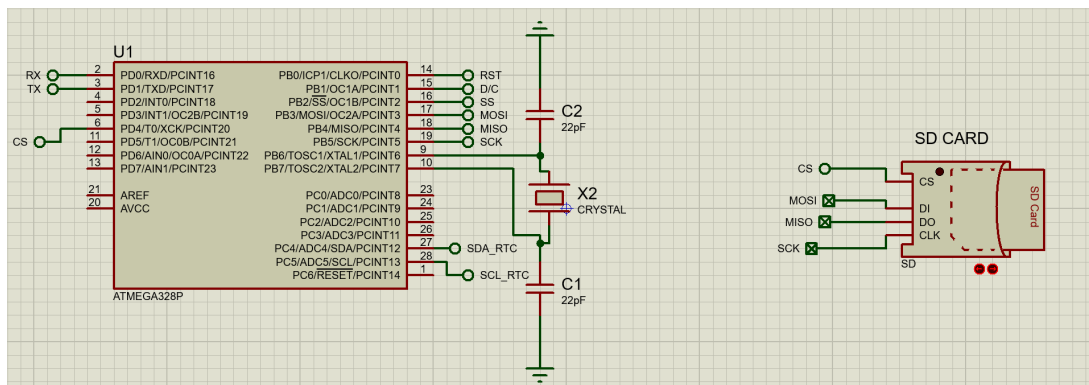


FIGURE 4.8 – Schéma de SD-CARD sous Proteus

4.5.3.1 Programme de SD card :

Pour utiliser le module de carte SD, nous devons inclure les bibliothèques nécessaires et initialiser la carte SD dans le code.

Voici quelques points clés de notre programme qui gèrent la carte SD et son contenu :

1. La fonction DisplayFileContent construit le nom de fichier en fonction d'un fingerprintID donné, en le convertissant en une chaîne au format "p<ID>.txt".
2. Elle tente ensuite d'ouvrir ce fichier en utilisant SD.open.
3. Si le fichier s'ouvre avec succès, la fonction lit le contenu du fichier.
4. Dans une boucle, la fonction vérifie s'il reste des caractères à lire en utilisant file.available().
5. Elle lit chaque caractère un par un en utilisant file.read(), les traite pour l'affichage, puis ferme le fichier en utilisant file.close() pour s'assurer que les ressources système sont correctement libérées.
6. Si le fichier ne peut pas être ouvert, indiquant qu'il n'existe pas ou qu'il y a une erreur, la fonction contourne le processus de lecture et gère l'erreur de manière appropriée.

4.5.4 L'Horloge RTC DS3231

L'horloge temps réel (RTC) DS3231 est un composant essentiel de notre système d'assistance intelligent, permettant de maintenir une mesure précise du temps même en cas de coupure de l'alimentation principale. Ce module utilise une interface de communication I2C

et intègre un oscillateur à cristal compensé en température (TCXO) pour garantir une précision exceptionnelle.

Pour connecter l'horloge RTC ds3231 à un microcontrôleur ATmega328P (par exemple, sur une carte Arduino Uno), suivez les connexions suivantes :

- **VCC** de l'horloge RTC DS3231 relié à la source d'alimentation de 3,3 V à 6 V.
- **GND** de l'horloge RTC DS3231 relié au GND du microcontrôleur.
- **SDA** de l'horloge RTC DS3231 reliée à la broche SDA (Digital Pin 27) du microcontrôleur.
- **SCL** de l'horloge RTC DS3231 relié à la broche SCL (Digital Pin 28) du microcontrôleur.

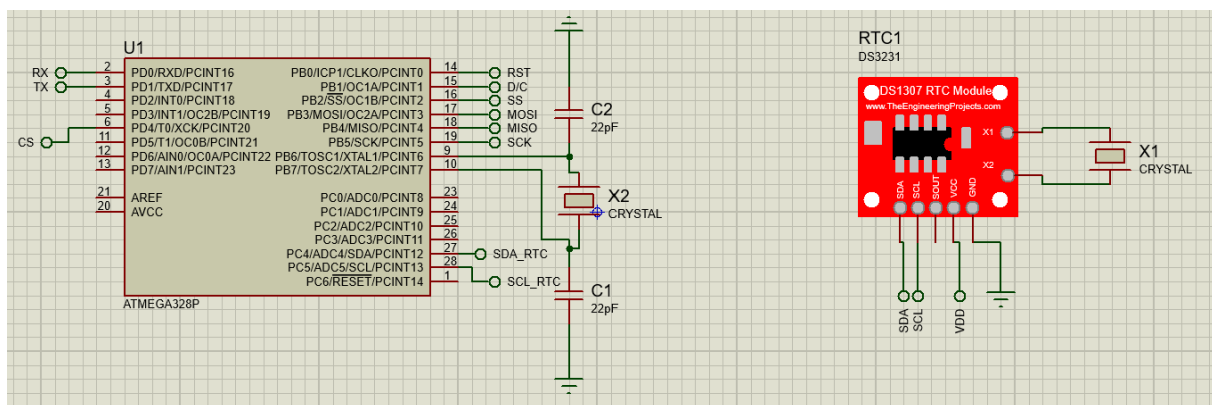


FIGURE 4.9 – Schéma de RTC DS3231 sous Proteus

4.5.4.1 Programme de L'Horloge RTC DS3231 :

Pour utiliser le module DS3231 avec Arduino, nous devons inclure les bibliothèques nécessaires et initialiser l'horloge dans le code. Ensuite, nous déclarons ces appels de fonction `RTC.getHours()`, `RTC.getMinutes()`, `RTC.getSeconds()`, `RTC.getDay()`, `RTC.getMonth()`, et `RTC.getYear()` -pour récupérer l'heure et la date précises actuelles du module RTC. Le module RTC maintient une mesure du temps précise même lorsque l'Arduino est éteint. Ces informations sont ensuite formatées en chaînes lisibles par l'homme pour être affichées à l'écran.

4.5.5 L'alimentation

Le système est alimenté par une tension de 5 V à l'aide d'un régulateur de 5 V et d'une batterie de 9 V.

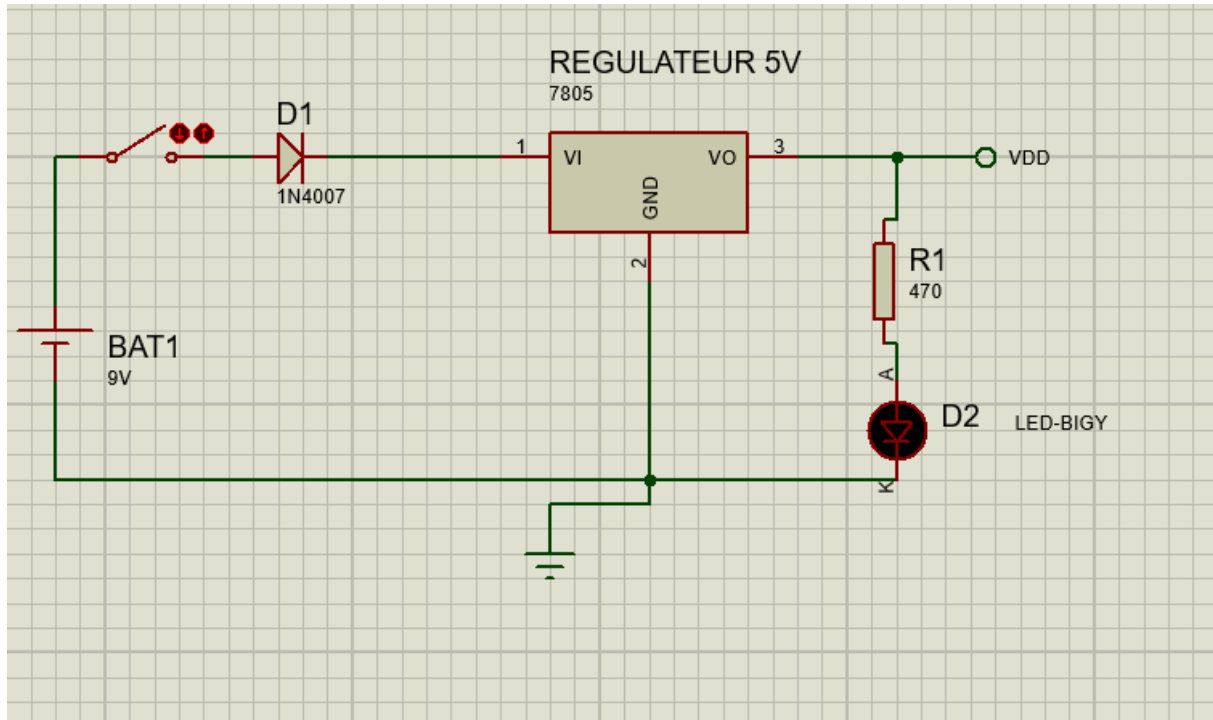


FIGURE 4.10 – Schéma de batterie avec régulateur sous Proteus

4.5.6 Le microcontrôleur

Le microcontrôleur ATmega328 est le cœur et l'âme du système, orchestrant toutes ses fonctions vitales. Ce composant essentiel joue un rôle crucial dans la collecte, le traitement et l'affichage des informations médicales cruciales, s'élevant en véritable chef d'orchestre de ce dispositif révolutionnaire.

Les responsabilités du microcontrôleur ATmega328 s'articulent autour de plusieurs axes fondamentaux :

1. **Maîtrise du flux d'informations :** le microcontrôleur agit comme un chef d'orchestre, gérant le flux de données entre les différents modules du système. Il reçoit les données biométriques du lecteur d'empreintes digitales FPM10A, récupère les informations

médicales stockées sur la carte SD et communique avec l'écran TFT pour afficher les données pertinentes.

2. **Traitement des données biométriques :** Une fois les empreintes digitales du patient capturées, le microcontrôleur entre en action, analysant et comparant les données biométriques aux informations stockées dans sa mémoire. Cette comparaison permet de vérifier l'identité du patient et d'autoriser ou non l'accès aux informations médicales.
3. **Obtention de la date et de l'heure actuelles :** Le microcontrôleur utilise l'horloge RTC DS3231 pour obtenir la date et l'heure actuelles.
4. **Affichage des informations médicales et de la date et de l'heure actuelles :** Si l'authentification par empreintes digitales est réussie, le microcontrôleur extrait les informations médicales pertinentes de la carte SD et les transmet à l'écran TFT pour un affichage clair et concis. Cela permet aux professionnels de santé d'accéder rapidement aux données vitales du patient, facilitant ainsi la prise de décision et les soins médicaux.
5. **Gestion du système :** Le microcontrôleur supervise l'ensemble du système, s'assurant que chaque module fonctionne de manière optimale. Il gère la communication entre les différents composants, contrôle l'alimentation électrique et veille au bon déroulement des opérations.

Voir l'organigramme de notre microcontrôleur :

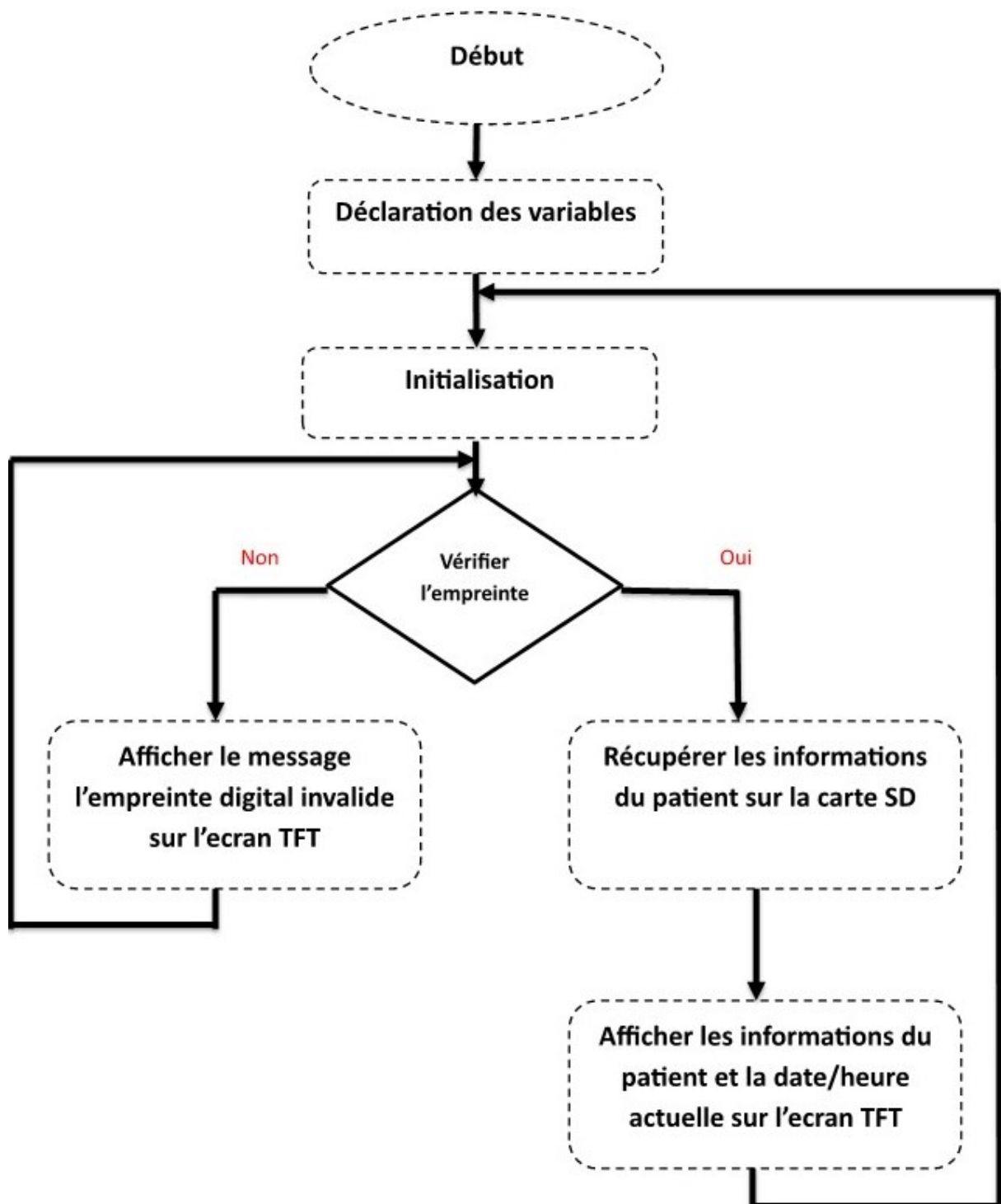


FIGURE 4.11 – L’organigramme global de microcontrôleur

4.6 Fonctionnement du système :

Le système d’assistance intelligent comprend plusieurs composants, dont un capteur d’empreintes digitales (FPM 10), un microcontrôleur ATmega328P, un écran TFT, une

horloge temps réel (RTC) et un bouton pour démarrer le système. Dans des conditions de fonctionnement normales, lorsque le système est activé, l'écran TFT affiche initialement « Scan prêt » après l'initialisation du capteur d'empreintes digitales.

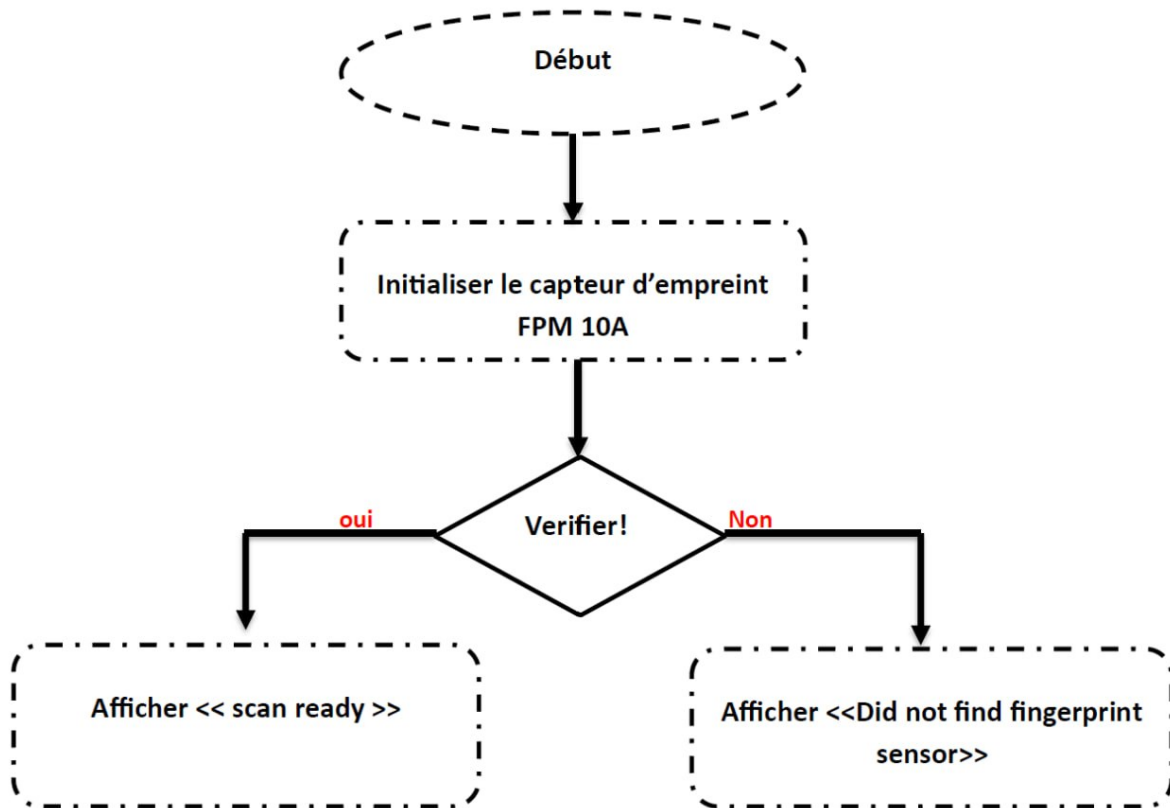


FIGURE 4.12 – L'organigramme de la partie d'initialisation du capteur FPM10A



FIGURE 4.13 – Image scan ready

Ensuite, lorsqu'une empreinte digitale est placée sur le capteur, le système vérifie si l'empreinte digitale correspond à l'un des modèles enregistrés dans la base de données du capteur FPM. Si l'empreinte digitale ne correspond à aucun modèle enregistré, l'écran affichera « RÉESSAYER ».

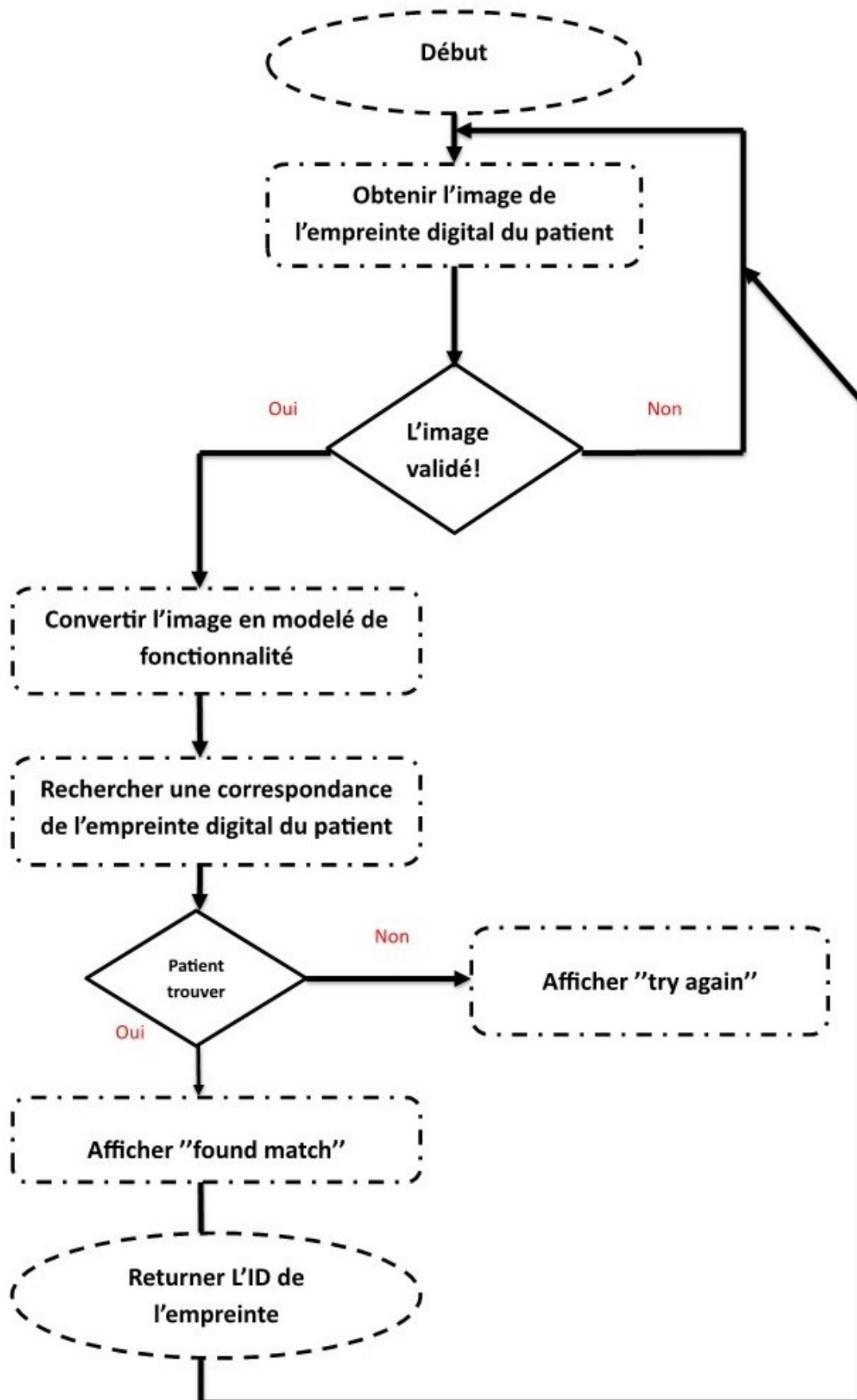


FIGURE 4.14 – L’organigramme de la partie d’obtention L’ID d’empreint digitale par FPM10A

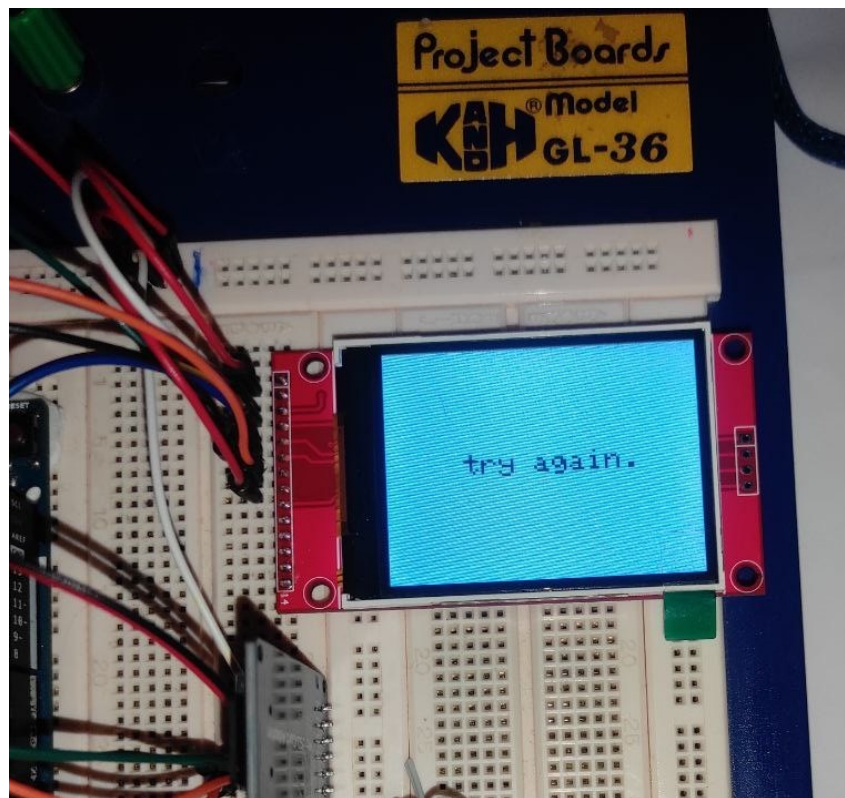


FIGURE 4.15 – Image Try again.

À l'inverse, si une correspondance est trouvée, l'écran indiquera « Correspondance trouvée » puis affichera les informations du fichier associées à l'identifiant d'empreinte digitale correspondant.

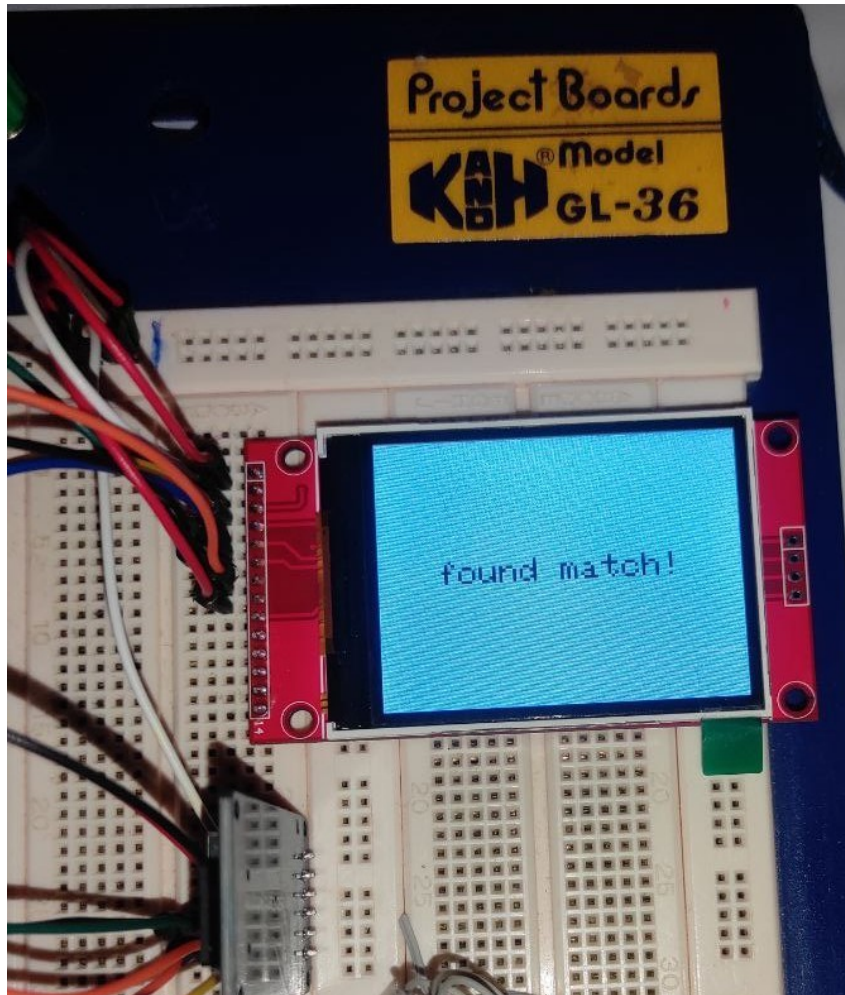


FIGURE 4.16 – Image found match

Après avoir affiché les informations du fichier de l’empreinte digitale correspondante pendant 20 secondes, le système affichera alors l’heure et la date actuelles correspondant au moment où la correspondance d’empreinte digitale s’est produite.

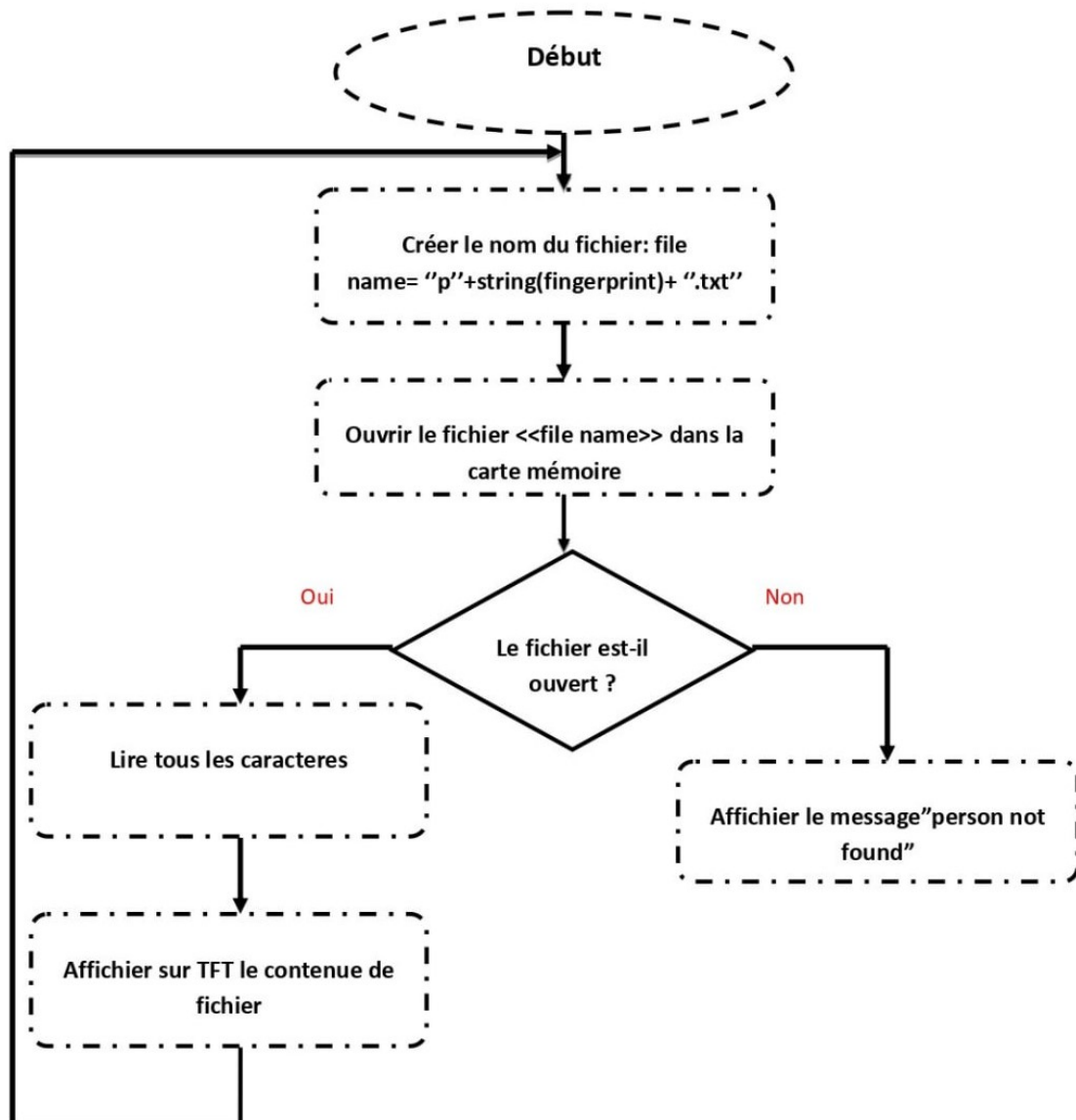


FIGURE 4.17 – L’organigramme de la partie du carte mémoire

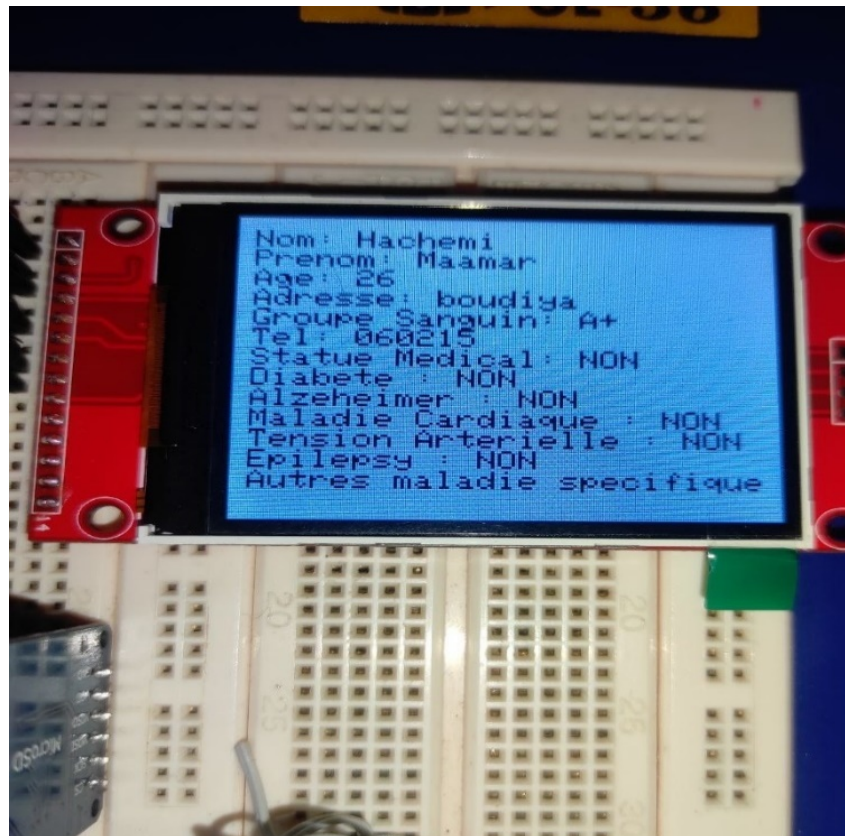


FIGURE 4.18 – Image, information affichée



FIGURE 4.19 – Image : la date et le temps

4.7 Essais :

Après avoir terminé notre PCB et l'avoir enfermé dans une boîte compacte, nous avons testé notre appareil en suivant les étapes suivantes pour nous assurer qu'il est fiable et efficace pour fournir des informations critiques sur une personne inconsciente dans des situations d'urgence médicale :



FIGURE 4.20 – Photo de la personne inconsciente avant de scanner ses informations



FIGURE 4.22 – Photo de l'appareil correspondant trouvé pour la victime



FIGURE 4.21 – Photo de l'appareil prêt pour le scan



FIGURE 4.23 – Photo des informations affichées après que la personne correspondante a été trouvée



FIGURE 4.24 – Photo d'après les informations sur l'heure est affichée

4.8 CONCLUSION

Ce chapitre présente en détail la mise en œuvre du prototype du système d'assistance intelligent. Les composants matériels tels que le capteur d'empreintes digitales, l'écran TFT, la carte SD et l'horloge RTC ont été connectés et interfacés avec le microcontrôleur. Des programmes spécifiques gèrent les fonctionnalités clés du système, comme l'enrôlement des empreintes digitales, l'affichage d'informations, le stockage de données. Ces composants ont été intégrés dans un programme unifié. Le système permet l'authentification biométrique, l'affichage d'informations, le stockage de données et la journalisation d'événements. Bien que certaines limitations existent, ce prototype démontre la faisabilité d'un tel système intelligent et son potentiel d'évolution future, ouvrant la voie à des solutions intelligentes dans les soins de santé et dans d'autres domaines.



CONCLUSION GÉNÉRALE

Les situations d'urgence médicale représentent des défis complexes, où une communication efficace et rapide est cruciale pour sauver des vies. Cette thèse a exploré les lacunes actuelles dans la transmission d'informations entre les patients, les équipes médicales et les services d'urgence, et a proposé une solution pratique sous la forme d'un système embarqué intelligent. Après avoir présenté le contexte des services d'aide médicale urgente et les principes de base du secourisme (Chapitre 1), nous avons examiné les capacités de la plateforme Arduino Uno et de son micro-contrôleur ATmega328P (Chapitre 2), ainsi que les composants électroniques complémentaires tels que le capteur d'empreintes digitales, l'écran TFT et le module de stockage (Chapitre 3). Ces éléments ont permis de concevoir et de prototyper un système embarqué capable d'identifier les patients, d'afficher leurs informations médicales et de stocker des données pertinentes (Chapitre 4). Ce prototype représente une contribution significative dans l'amélioration de la gestion des urgences médicales. En fournissant un accès rapide aux informations cruciales du patient, même lorsque celui-ci est inconscient, il permet aux équipes d'intervention de prendre des décisions éclairées et de prodiguer les soins appropriés sans délai. De plus, la capacité de stocker des données médicales actualisées facilite le suivi et la continuité des soins. Bien que fonctionnel, ce système offre de nombreuses perspectives d'amélioration. L'intégration de capteurs supplémentaires pour surveiller les signes vitaux, l'ajout d'une connexion sans fil pour transmettre les données en temps réel et l'optimisation de l'interface utilisateur sont autant de pistes à explorer. De plus, une étude approfondie de l'acceptabilité et de l'utilisation de ce système par les professionnels de la santé et les patients serait bénéfique pour en assurer l'adoption et l'efficacité sur le terrain. En somme, cette thèse a permis de relever les défis de communication dans les situations d'urgence médicale en proposant une solution technologique innovante. Grâce à une approche interdisciplinaire

combinant les concepts de secourisme, d'électronique embarquée et de conception de systèmes, elle ouvre la voie à de nouvelles avancées pour optimiser la prise en charge des patients et sauver des vies précieuses.



Bibliographie

- [1] Ministère des Solidarités et de la Santé. Le samu. URL <https://solidarites-sante.gouv.fr>. Accessed : 2024-06-04.
- [2] National SAMU-Urgences de France. Samu-urgences de france. URL <https://www.samu-urgences-de-france.fr>. Accessed : 2024-06-04.
- [3] Cour des comptes. Le service d'aide médicale urgente (samu) et le service mobile d'urgence et de réanimation (smur), oct 2023. URL https://www.ccomptes.fr/sites/default/files/2023-10/20230524-RALFSS-2023_0.pdf. Consulté le 4 juin 2024.
- [4] Société Française de Médecine d'Urgence. La cour des comptes plaide pour une plus grande adaptation des types d'intervention de l'aide médicale urgente. URL https://www.sfm.org/fr/actualites/actualites-de-l-urgences/la-cour-des-comptes-plaide-pour-une-plus-grande-adaptation-des-types-d-intervention-de-l-aide-medicale-urgente/new_id/69098. Accessed : 2024-06-04.
- [5] Lina Aboueljinane. *Evaluation et amélioration des performances des Systèmes d'Aide Médicale Urgente : application au SAMU du département du Val de Marne*. Theses, Ecole Centrale Paris, June 2014. URL <https://theses.hal.science/tel-01089138>.
- [6] Besma Glaa. *CONTRIBUTION A LA CONCEPTION ET L'OPTIMISATION D'UN SYSTEME D'AIDE A LA GESTION DES URGENCES*. Theses, Ecole Centrale de Lille, December 2008. URL <https://theses.hal.science/tel-00359607>.
- [7] m-soigner.com. Les nouvelles technologies investissent le samu. URL <https://www.m-soigner.com/pratiques/sant%C3%A9-num%C3%A9rique/485-les-nouvelles-technologies-investissent-le-samu-44.html>. Accessed : 2024-06-04.
- [8] MSD Connect. L'innovation technologique médicale au service de la santé. URL <https://www.msconnect.fr/innovation-sante/e-sante/linnovation-technologique-medecale-au-service-de-la-sante/>. Accessed : 2024-06-04.
- [9] FasterCapital. Soins d'urgence : Innovations en matière de soins d'urgence - comment les startups révolutionnent la réponse médicale. URL <https://fastercapital.com/fr/contenu/Soins-d-urgence-Innovations-en-matiere-de-soins-d-urgence---comment-les-startups-revolutionnent-la-reponse-medecale.html>. Accessed : 2024-06-04.
- [10] Jean Marty. Les défis de l'aide médicale d'urgence. *Soins*, 58(777) :40, July 2013. ISSN 0038-0814. doi : 10.1016/j.soin.2013.04.034.
- [11] Larousse. *Petit Larousse Illustré*. Éditions Larousse, 2023. Article sur le secourisme.

- [12] Institut National de Prévention et d'Éducation pour la Santé (INPES). *Les Premiers Secours en Cas d'Accident*. INPES, 2012.
- [13] Article 434-1 du code pénal. *Code pénal*, 2023. URL https://www.legifrance.gouv.fr/codes/article_lc/LEGIARTI000006417893/.
- [14] Jean-Louis Vincent and Jacques Berré. *Le manuel de réanimation, soins intensifs et médecine d'urgence*. Springer, 2009.
- [15] Croix-Rouge française. *Premiers secours : Gestes qui sauvent*. Éditions Nathan, 2020. Ce livre fournit des instructions détaillées sur les gestes de premiers secours, incluant les techniques pour évaluer l'état de conscience d'une victime en posant des questions simples, en donnant des ordres simples et en utilisant des stimulations physiques comme un pincement léger.
- [16] Manuel Jiménez, Rogelio Palomera, and Isidoro Couvertier. *Introduction to embedded systems*. Springer, 2013.
- [17] Techno-Science.net. Microcontrôleur, 2024. URL <https://www.techno-science.net/glossaire-definition/Microcontroleur.html>. Accessed : 2024-06-14.
- [18] John L Hennessy and David A Patterson. *Computer architecture : a quantitative approach*. Elsevier, 2011.
- [19] David A. Patterson and John L. Hennessy. *Computer Organization and Design : The Hardware/Software Interface*. Morgan Kaufmann, San Francisco, 5th edition, 2013. ISBN 9780124077263.
- [20] Tim Wilmshurst. *Designing Embedded Systems with PIC Microcontrollers : Principles and Applications*. Newnes, Oxford, 2nd edition, 2009. ISBN 9781856177504.
- [21] Muhammad Ali Mazidi, Sepehr Naimi, and Sarmad Naimi. *The AVR microcontroller and embedded systems : using Assembly and C*. MicroDigitalEd. com, 2017.
- [22] Dogan Ibrahim. *Advanced PIC microcontroller projects in C : from USB to RTOS with the PIC 18F Series*. Newnes, 2011.
- [23] Kenneth J. Ayala. *The 8051 Microcontroller*. Thomson Delmar Learning, 3rd edition, 2005. ISBN 978-1401861582.
- [24] Patrice Kadionik. *Le Microcontrôleur 8 Bits 68HC11*, 2004. Available at : www.enseirb.fr/~kadionik.
- [25] Ioan Susnea and Marian Mitescu. *Microcontrollers in Practice*. Springer, Berlin, Heidelberg, 2006. ISBN 978-3-540-29329-9.
- [26] Technologue Pro. Les microcontrôleurs pic, n.d. URL <https://www.technologuepro.com/microcontroleur/les-microcontroleurs-pic.htm>. Consulté le 17 juin 2024.
- [27] Elliot Williams. *Make : AVR programming*. Shroff Publishers & Distr, 2014.
- [28] M Banzi. *Arduino programming in 24 hours, sams teach yourself*, 2023.

- [29] Atmel Corporation. *ATmega328/P : 8-bit AVR Microcontroller with 32K Bytes In-System Programmable Flash*, 2014. URL http://www.atmel.com/Images/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf. Datasheet.
- [30] Microchip Technology Inc. *ATmega328P Data Sheet*, 2018. URL https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf. Accessed : 2024-06-18.
- [31] Robert Graybill and Rami Melhem. *Power aware computing*. Springer Science & Business Media, 2013.
- [32] Richard H. Barnett, Sarah Cox, and Larry O’Cull. *Embedded C Programming and the Atmel AVR*. Cengage Learning, 2010. ISBN 978-1439806104. Provides a comprehensive overview of programming and using the ATmega328P, including sections on power management.
- [33] John Nussey. *Arduino for dummies*. John Wiley & Sons, 2013.
- [34] Simon Monk. *Programming Arduino : Getting Started with Sketches*. McGraw-Hill Education TAB, 2 edition, 2017.
- [35] Michael Margolis, Brian Jepson, and Nicholas Robert Weldin. *Arduino Cookbook*. O’Reilly Media, Inc., 3rd edition, April 2020. ISBN 9781491903520.
- [36] Hakan YUKSEL. The lilypad arduino. *Programmable Smart Microcontroller Cards*, page 163, 2021.
- [37] Jeremy Blum. *Exploring Arduino : Tools and Techniques for Engineering Wizardry*. John Wiley & Sons, 2013.
- [38] Hillary Moses Daluz. *Fingerprint analysis laboratory workbook*. CRC Press, 2018.
- [39] Davide Maltoni, Dario Maio, Anil K Jain, Salil Prabhakar, et al. *Handbook of fingerprint recognition*, volume 2. Springer, 2009.
- [40] OUKIL Kahina and ZERBOUT Ania. *Contrôle d’accès à base d’empreinte digitale*. PhD thesis, Université Mouloud Mammeri, 2019.
- [41] Stan Z Li and Anil Jain. *Encyclopedia of biometrics*. Springer Publishing Company, Incorporated, 2015.
- [42] Anil K Jain, Patrick Flynn, and Arun A Ross. *Handbook of biometrics*. Springer Science & Business Media, 2007.
- [43] Arokia Nathan, Arman Ahnood, Matthew T Cole, Sungsik Lee, Yuji Suzuki, Pritesh Hiralal, Francesco Bonaccorso, Tawfique Hasan, Luis Garcia-Gancedo, Andriy Dyadyusha, et al. Flexible electronics : the next ubiquitous platform. *Proceedings of the IEEE*, 100(Special Centennial Issue) :1486–1517, 2012.
- [44] James L Wayman, Anil K Jain, Davide Maltoni, and Dario Maio. *Biometric systems : Technology, design and performance evaluation*. Springer Science & Business Media, 2005.

- [45] Tsai-Yang Jea and Venu Govindaraju. A minutia-based partial fingerprint recognition system. *Pattern recognition*, 38(10) :1672–1684, 2005.
- [46] Jiun-Haw Lee, I-Chun Cheng, Hong Hua, and Shin-Tson Wu. *Introduction to flat panel displays*. John Wiley & Sons, 2020.
- [47] Janglin Chen, Wayne Cranton, and Mark Fihn. *Handbook of visual display technology*. Springer Publishing Company, Incorporated, 2011.
- [48] Robert H Chen. *Liquid crystal displays : fundamental physics and technology*. John Wiley & Sons, 2011.
- [49] Ernst Lueder, Peter Knoll, and Seung Hee Lee. *Liquid crystal displays : addressing schemes and electro-optical effects*. John Wiley & Sons, 2022.
- [50] Orient Display. Amoled vs ips tft display – which is better display technology?, 2024. URL <https://www.orientdisplay.com/fr/knowledge-base/lcd-basics/amoled-vs-ips-tft-display-which-%20is-better-display-technology/>.
- [51] Ronald Sass and Andrew G Schmidt. *Embedded systems design with platform FPGAs : principles and practices*. Morgan Kaufmann, 2010.
- [52] Steven F Barrett. *Arduino VI : Bioinstrumentation*. Springer Nature, 2023.
- [53] Neil Cameron, Neil Cameron, and Pao. *Arduino Applied*. Springer, 2019.

Annexe A

Listing du programme complet

```
1  #include <Adafruit_GFX.h> // Bibliothèque pour les fonctions graphiques
2  //génériques
3  #include <Adafruit_ST7789.h> // Bibliothèque pour l'écran LCD ST7789
4  #include <SPI.h> // Bibliothèque SPI pour la communication série
5  #include <SD.h> // Bibliothèque pour la carte SD
6  #include <Wire.h> // Bibliothèque I2C pour la communication série
7  #include <I2C_RTC.h> // Bibliothèque pour l'horloge temps réel (RTC)
8
9  static DS3231 RTC; // Objet RTC pour interagir avec l'horloge temps réel
10 #include <Adafruit_Fingerprint.h> // Bibliothèque pour le capteur //
11 //d'empreintes digitales
12
13 Adafruit_Fingerprint finger = Adafruit_Fingerprint(&Serial);
14 int fingerprintID = 0; // Variable pour stocker l'ID de l'empreinte digitale
15
16 #define TFT_CS 10 // Broche de sélection du circuit pour le TFT
17 #define TFT_DC 9 // Broche de données/commande pour le TFT
18 #define TFT_RST 8 // Broche de réinitialisation pour le TFT
19
20 // Initialiser l'objet d'affichage TFT
21 Adafruit_ST7789 tft = Adafruit_ST7789(TFT_CS, TFT_DC, TFT_RST);
22
```

```
23  const int chipSelect = 4; // Broche de sélection du circuit pour la carte SD
24
25  #define DISPLAY_WIDTH 320 // Largeur de l'écran
26  #define DISPLAY_HEIGHT 240 // Hauteur de l'écran
27  #define HORIZONTAL_MARGIN 10 // Marge horizontale
28  #define VERTICAL_MARGIN 10 // Marge verticale
29  #define TIME_AREA_HEIGHT 18 // Hauteur de la zone d'affichage du temps
30
31  // Fonction pour initialiser le RTC
32  void initRTC3231() {
33      RTC.begin(); // Démarrer le RTC
34
35      // Vérifier si l'horloge fonctionne et afficher le statut
36      Serial.print(F("L'horloge fonctionne-t-elle : "));
37      if (RTC.isRunning())
38          Serial.println("Oui");
39      else
40          Serial.println(F("Non. L'heure peut être inexacte"));
41
42      // Vérifier et afficher le mode de l'heure (12 heures ou 24 heures)
43      Serial.print(F("Mode heure : "));
44      if (RTC.getHourMode() == CLOCK_H24)
45          Serial.println(F("24 Heures"));
46      else
47          Serial.println(F("12 Heures"));
48
49      // Vérifier et afficher le statut de l'alarme 1
50      Serial.print("L'alarme 1 est-elle activée : ");
51      if (RTC.isAlarm1Enabled())
52          Serial.println(F("Oui"));
53      else
```

```
54     Serial.println(F("Non"));
55 }
56
57 // Fonction pour initialiser la carte SD
58 void initSDCard() {
59     if (!SD.begin(chipSelect)) {
60         Serial.println(F("Échec de l'initialisation de la carte SD !"));
61     } else {
62         Serial.println(F("Initialisation de la carte SD terminée.));
63         displayMessageCentered("Initialisation de la carte SD terminée.));
64     }
65 }
66
67 void setup() {
68     Serial.begin(9600); // Initialiser la communication série
69     tft.init(240, 320, SPI_MODE0); // Initialiser l'écran TFT
70     tft.fillScreen(ST77XX_BLACK); // Effacer l'écran TFT
71     initSDCard(); // Initialiser la carte SD
72     initRTC3231(); // Initialiser le RTC
73
74     finger.begin(57600); // Initialiser le capteur d'empreintes digitales
75
76     tft.setRotation(1); // Définir la rotation de l'affichage
77     tft.setTextWrap(false); // Désactiver le retour à la ligne automatique
78     tft.setCursor(HORIZONTAL_MARGIN, VERTICAL_MARGIN + 100);
79     // Définir la position du curseur
80
81     tft.setTextColor(ST77XX_YELLOW, ST77XX_BLACK); // Définir la couleur du texte
82     tft.setTextSize(2); // Définir la taille du texte
83     tft.print("Projet NAJDATY\n PAR :NABI && HACHEMI");
84     // Afficher le nom du projet et des auteurs
```

```
85
86 delay(4000); // Attendre 4 secondes
87
88 // Vérifier si le capteur d'empreintes digitales est trouvé
89 if (finger.verifyPassword()) {
90     Serial.println(F("Capteur d'empreintes digitales trouvé !"));
91     displayMessageCentered("scan ready");
92 } else {
93     Serial.println(F("Capteur d'empreintes digitales non trouvé :("));
94     displayMessageCentered("Capteur d'empreintes digitales non trouvé");
95 }
96 }
97
98 void loop() {
99     fingerprintID = getFingerprintIDez(); // Obtenir l'ID de l'empreinte digitale
100     delay(50); // Attendre 50 millisecondes
101
102     if (fingerprintID != -1) {
103         displayFileContent(fingerprintID);
104         // Afficher le contenu en fonction de l'ID de l'empreinte digitale
105         delay(4000); // Attendre 4 secondes
106         displayTimeDate(); // Afficher l'heure et la date actuelles
107     }
108 }
109
110
111 // Fonction pour obtenir l'ID de l'empreinte digitale
112 int getFingerprintIDez() {
113     uint8_t p = finger.getImage(); // Capturer l'image de l'empreinte digitale
114     if (p != FINGERPRINT_OK) return -1; // Retourner -1 si la capture d'image a échoué
115 }
```

```
116 p = finger.image2Tz(); // Convertir l'image en modèle d'empreinte digitale
117 if (p != FINGERPRINT_OK) return -1; // Retourner -1 si la conversion a échoué
118
119 p = finger.fingerFastSearch();
120 // Rechercher dans la base de données d'empreintes digitales
121 if (p != FINGERPRINT_OK) {
122     displayMessageCentered("Try again.");
123     return -1; // Retourner -1 si la recherche a échoué
124 }
125
126 displayMessageCentered("match founde !"); // Afficher le message de
127 // correspondance trouvée
128 Serial.print(finger.fingerID); // Afficher l'ID de l'empreinte digitale
129
130 delay(3000); // Attendre 3 secondes
131
132 return finger.fingerID; // Retourner l'ID de l'empreinte digitale
133 }
134
135 int timeAreaX = 0; // Position X de la zone de temps
136 int timeAreaY = 0; // Position Y de la zone de temps
137 int timeAreaWidth = DISPLAY_WIDTH; // Largeur de la zone de temps
138 int timeAreaHeight = TIME_AREA_HEIGHT; // Hauteur de la zone de temps
139
140 // Fonction pour afficher le contenu du fichier en fonction de l'ID de
141 // l'empreinte digitale
142 void displayFileContent(int fingerprintID) {
143     String fileName = "p" + String(fingerprintID) + ".txt"; // Construire le
144     // nom du fichier
145     File file = SD.open(fileName); // Ouvrir le fichier depuis la carte SD
146
```

```
147 if (file) {
148     tft.fillScreen(ST77XX_BLACK); // Effacer l'écran TFT
149     tft.setCursor(HORIZONTAL_MARGIN, VERTICAL_MARGIN); // Définir
150
151     // la position du curseur
152     tft.setTextColor(ST77XX_WHITE, ST77XX_BLACK); // Définir la couleur du texte
153     tft.setTextSize(2); // Définir la taille du texte
154
155     while (file.available()) {
156         char c = file.read(); // Lire un caractère depuis le fichier
157
158         if (c == '\n') {
159             tft.setCursor(HORIZONTAL_MARGIN, tft.getCursorY() + 16); // Passer à la li
160             // suivante
161         } else {
162             tft.print(c); // Afficher le caractère
163         }
164     }
165
166     file.close(); // Fermer le fichier
167 } else {
168     displayMessageCentered("personne non trouvée"); // Afficher un
169     // message si le fichier n'est pas trouvé
170 }
171
172
173 // Fonction pour afficher un message centré sur l'écran TFT
174 void displayMessageCentered(const char* message) {
175     uint16_t textWidth = tft.width();
176     uint16_t textHeight = tft.height();
177     int16_t x, y;
```

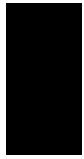
```
178
179 tft.getTextBounds(message, 0, 0, &x, &y, &textWidth, &textHeight); // Obtenir le
180
181 x = (tft.width() - textWidth) / 2; // Calculer la position x centrée
182 y = (tft.height() - textHeight) / 2; // Calculer la position y centrée
183 tft.fillScreen(ST77XX_BLACK); // Effacer l'écran TFT
184 tft.setCursor(x, y); // Définir le curseur sur la position centrée
185 tft.setTextColor(ST77XX_WHITE); // Définir la couleur du texte
186 tft.setTextSize(3); // Définir la taille du texte
187
188 tft.println(message); // Afficher le message
189 }
190
191 // Fonction pour afficher l'heure et la date actuelles sur l'écran TFT
192 void displayTimeDate() {
193     tft.fillScreen(ST77XX_BLACK); // Effacer l'écran TFT
194     tft.setCursor(HORIZONTAL_MARGIN + 70, 100); // Définir la position //du curseur
195     tft.setTextColor(ST77XX_GREEN, ST77XX_BLACK); // Définir la //couleur du texte
196     tft.setTextSize(3); // Définir la taille du texte
197
198     char timeStr[9];
199     char dateStr[11];
200
201     // Formater l'heure et la date actuelles
202     snprintf(timeStr, sizeof(timeStr), "%02d:%02d:%02d"
203 , RTC.getHours(), RTC.getMinutes(), RTC.getSeconds());
204     snprintf(dateStr, sizeof(dateStr), "%02d/%02d/%04d",
205 RTC.getDay(), RTC.getMonth(), RTC.getYear());
206
207     tft.print(timeStr); // Afficher l'heure actuelle
208     tft.setCursor(HORIZONTAL_MARGIN + 70, 100 + TIME_AREA_HEIGHT); // Définir la
```

```
209 //position du curseur pour la date
210 tft.println(dateStr); // Afficher la date actuelle
211 }
```

Websites (sites Internet)

tutoriel sur la façon d'inscrire des empreintes digitales :

<https://randomnerdtutorials.com/fingerprint-sensor-module-with-arduino/>



Annexe B

Datasheet de Module TFT ST7789 :

https://www.waveshare.com/wiki/File:ST7789_Datasheet.pdf



ST7789VW

Datasheet

Sitronix reserves the right to change the contents in this document without prior notice, please contact Sitronix to obtain the latest version of datasheet before placing your order. No responsibility is assumed by Sitronix for any infringement of patent or other rights of third parties which may result from its use.

© 2017 Sitronix Technology Corporation. All rights reserved.

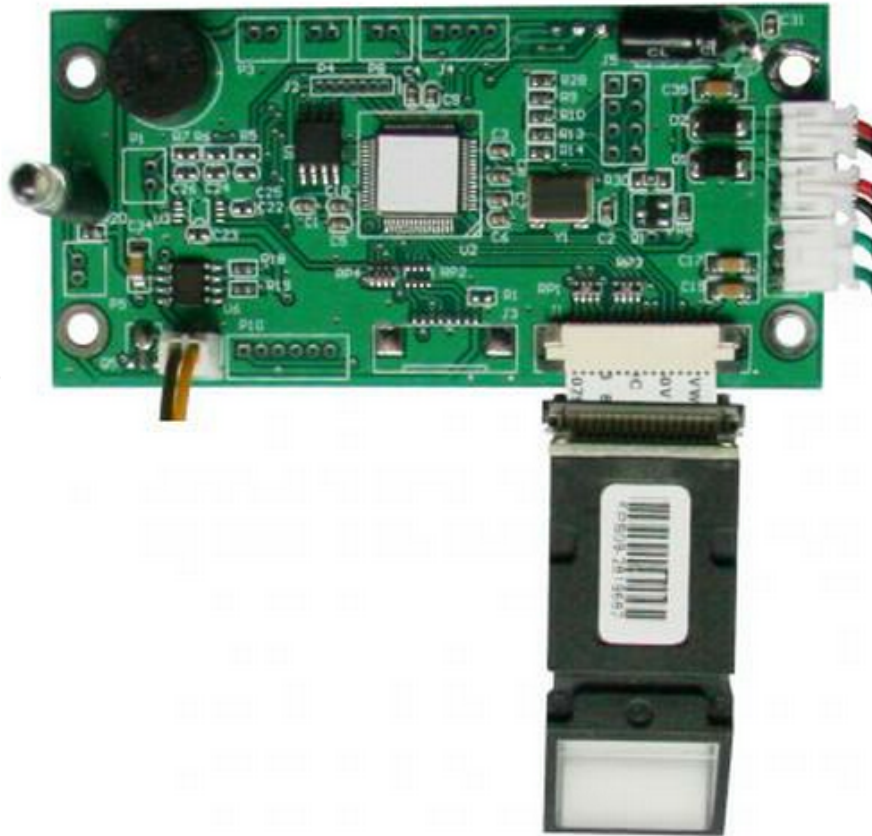
Version 1.0

2017/09

Datasheet de Module FPM 10A :

<https://dokumen.tips/documents/finger-print-fpm-10a-bio-module-datasheet.html>

Bio Module FPM-10



FPM-10 series Fingerprint Module is one of our high-end biometric products. With patented fingerprint algorithm, the fingerprint module solves the problem for dry fingers. Composed by optical fingerprint sensor and a PCB with a large-capacity chip, the modules can be applied for door locks, safes or any controllers.

With our best technology and quality control, our fingerprint technology has been applied to different markets on safes, vaults, deposit boxes and other applications in North America.

Datasheet de Module RTC DS3231 :

<https://www.analog.com/media/en/technical-documentation/data-sheets/ds3231.pdf>

DS3231

Extremely Accurate I²C-Integrated RTC/TCXO/Crystal

General Description

The DS3231 is a low-cost, extremely accurate I²C real-time clock (RTC) with an integrated temperature-compensated crystal oscillator (TCXO) and crystal. The device incorporates a battery input, and maintains accurate timekeeping when main power to the device is interrupted. The integration of the crystal resonator enhances the long-term accuracy of the device as well as reduces the piece-part count in a manufacturing line. The DS3231 is available in commercial and industrial temperature ranges, and is offered in a 16-pin, 300-mil SO package.

The RTC maintains seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator. Two programmable time-of-day alarms and a programmable square-wave output are provided. Address and data are transferred serially through an I²C bidirectional bus.

A precision temperature-compensated voltage reference and comparator circuit monitors the status of V_{CC} to detect power failures, to provide a reset output, and to automatically switch to the backup supply when necessary. Additionally, the $\overline{\text{RST}}$ pin is monitored as a pushbutton input for generating a μP reset.

Benefits and Features

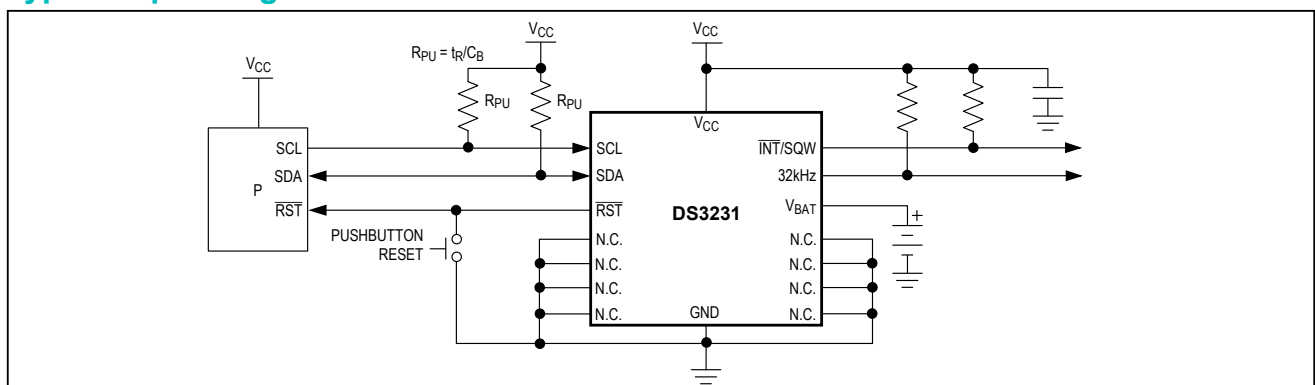
- Highly Accurate RTC Completely Manages All Timekeeping Functions
 - Real-Time Clock Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the Week, and Year, with Leap-Year Compensation Valid Up to 2100
 - Accuracy $\pm 2\text{ppm}$ from 0°C to +40°C
 - Accuracy $\pm 3.5\text{ppm}$ from -40°C to +85°C
 - Digital Temp Sensor Output: $\pm 3^\circ\text{C}$ Accuracy
 - Register for Aging Trim
 - $\overline{\text{RST}}$ Output/Pushbutton Reset Debounce Input
 - Two Time-of-Day Alarms
 - Programmable Square-Wave Output Signal
- Simple Serial Interface Connects to Most Microcontrollers
 - Fast (400kHz) I²C Interface
- Battery-Backup Input for Continuous Timekeeping
 - Low Power Operation Extends Battery-Backup Run Time
 - 3.3V Operation
- Operating Temperature Ranges: Commercial (0°C to +70°C) and Industrial (-40°C to +85°C)
- Underwriters Laboratories® (UL) Recognized

Applications

- Servers
- Telematics
- Utility Power Meters
- GPS

Ordering Information and Pin Configuration appear at end of data sheet.

Typical Operating Circuit



Underwriters Laboratories is a registered certification mark of Underwriters Laboratories Inc.