

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 أبي بكر بلقايد - تلمسان

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE



MEMOIRE

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

En : Electronique

Spécialité : Instrumentation

Par : HAMDANI ANES & LAHCENE OUSSAMA

Sujet

Conception et réalisation d'un affichage d'horloge , calendrier et heure de prière au moyen des coordonnées géographiques GPS

Soutenu publiquement, le 29/06/2025, devant le jury composé de :

M Kerai Nabil	MAA	Université de Tlemcen	Président
M Boursali Amine	MCB	Université de Tlemcen	Examineur
M Maasoum Nouredine	MCB	Université de Tlemcen	Encadreur
M Lachachi Djamel	MCB	Université de Tlemcen	Co-Encadreur

Année universitaire : 2024 / 2025

Remerciements

Tout d'abord, nous remercions Dieu tout-puissant pour son guidance et aide qui nous ont permis de mener à bien cet humble projet a terme.

Ce projet a eu lieu à l'université Abou Bekr Belkaid de Tlemcen, plus spécifiquement au sein du département de génie électrique et électronique de la faculté des technologies.

Je tiens à exprimer ma sincère gratitude à toutes les personnes qui ont contribué à la réussite de ce projet.

Tout d'abord, nous présentons notre sincère gratitude à notre encadreur, Mr MAASOUM NOUREDINE, pour avoir accepté de nous accompagner tout au long de ce travail. Avec sa disponibilité constante et ses conseils précis il nous a permis de mener à bien notre projet dans les meilleures conditions dès le début de sa préparation.

Je remercie également monsieur Lachachi Djamel, dont les échanges, les discussions et l'engagement ont enrichi nos réflexions et apporté des solutions aux défis rencontrés. Sa collaboration a grandement facilité la conception et la mise en œuvre des différentes étapes du système.

Nous ne saurions oublier nos familles et amis, dont le soutien constant, les encouragements et la patience ont été une source précieuse de motivation. Leur présence nous a permis de surmonter les défis et d'aborder cette expérience avec détermination.

Nous tenons à remercier, aussi, monsieur Kerai Nabil de nous avoir fait l'honneur, de bien vouloir accepter de présider ce jury.

Les remerciements sont, également, adressés à monsieur Boursali Amine Nous sommes très sensibles à l'honneur que vous nous faites en acceptant d'examiner le présent travail.

Ce projet est le fruit d'un travail collectif, et c'est grâce à ces contributions que son aboutissement a été possible. À tous, merci infiniment !

Dédicace

Je tiens à exprimer ma profonde reconnaissance à Dieu, source inépuisable de sagesse et de force, qui m'a guidé tout au long de cette aventure. Sa bienveillance et Son soutien m'ont permis de surmonter chaque obstacle et d'avancer avec confiance.

Je remercie également mon encadreur, Mr. MAASOUM NOUREDINE, dont l'expertise, les conseils avisés et l'accompagnement attentif ont été essentiels pour la réussite de ce projet. Grâce à son encadrement, j'ai pu affiner mes compétences et aborder chaque défi avec assurance.

À ma mère qui m'a soutenu, guidé et encouragé tout au long de la préparation de ce mémoire.

À mon père, dont l'aide précieuse et les conseils avisés m'ont permis d'aller au bout de mes idées.

À ma famille Hamdani, qui a toujours cru en moi, je dédie ce travail. Leur soutien constant, leur patience et leurs encouragements ont constitué une source inestimable de motivation et de réconfort tout au long de ce parcours.

Je tiens aussi à remercier mes amis, dont l'amitié sincère, la bonne humeur et la présence rassurante ont rendu cette aventure plus enrichissante et agréable.

Je remercie sincèrement mes professeurs pour leur enseignement et leur accompagnement, qui ont été essentiels à mon apprentissage et à la réussite de ce projet.

Enfin, à toutes les personnes qui, de près ou de loin, ont contribué à l'aboutissement de ce projet, j'exprime ma plus sincère gratitude.

HAMDANI ANES

Dédicace

Avant toute chose, je remercie Dieu de m'avoir donné la force et la persévérance indispensables pour mener à bien ce projet. Sa guidance m'a éclairé à chaque étape, m'inspirant à dépasser mes limites.

Je souhaite exprimer ma profonde gratitude à monsieur MAASOUM NOUREDINE, mon encadreur, dont les conseils éclairés et l'attention portée à mes idées ont grandement contribué à mon développement. Son soutien professionnel m'a permis de progresser de manière significative.

À la perle précieuse de mon existence, à ma chère mère qui a enduré sans me faire subir de souffrance, qui n'a jamais abandonné de me soutenir et d'être là pour m'aider à réaliser mon but. Ma mère, que Dieu la protège, n'a jamais refusé de répondre à mes demandes.

Je remercie sincèrement mon père pour son soutien constant et ses encouragements, essentiels à la réussite de ce projet.

À mes sœurs et ma famille, je dédie ce travail en signe de reconnaissance pour leur soutien inébranlable, leur patience et les encouragements constants qui m'ont soutenu lors des moments difficiles. Leur présence a été le pilier de ma motivation.

Je remercie également mes amis, dont l'esprit positif et la camaraderie ont transformé les défis en expériences enrichissantes et mes proches qui étaient avec moi lorsque ma bougie s'est éteinte avec leur soutien et leurs encouragements.

A tous mes chers professeurs, qui n'ont pas hésité à me tendre la main.

Enfin, à toutes les personnes ayant contribué, de près ou de loin, à la réussite de ce projet, je vous adresse mes remerciements les plus sincères.

LAHCENE OUSSAMA

Liste des abréviations

BCD: Binary Coded Decimal

ROM : Read Only Memory (Mémoire morte)

EPROM : Erasable Programmable Read Only Memory (ça veut dire mémoire morte)

GPS : Global Positioning System (Système mondial de positionnement)

LCD : Liquid Crystal Display (Dispositif d'affichage à cristaux liquides)

LED : Light-emitting diode (Diode électroluminescente)

OLED : Organic Light Emitting Diode (Diode organique électroluminescente)

RAM : Random Access Memory Mémoire vive)

I2C : Inter-Integrated Circuit (Protocole de communication)

UART : Universal Asynchronous Receiver/Transmitter (Protocole de communication)

SPI : Serial Peripheral Interface (Protocole de communication)

RTC : Real-time Clock (Horloge temps reel)

PCB : Printed Circuit Board (Circuit imprimé, polychlorobiphényle)

UTC : Coordinated Universal Time

GMT : Greenwich Mean Time

MWL : Muslim World League

ISNA : Islamic Society of North America

TZ : Time Zone

USB : Universal Serial Bus

UOIF : Union of Islamic Organisations of France

TFT : Thin Film Transistor

Table des figures

Figure 1 : Structure de l'écran LCD.....	16
Figure 2 : Configuration d'un afficheur 7 segments	18
Figure 3 : Brochage d'un circuit DS1307	21
Figure 4 : Brochage d'un circuit 74HC595	21
Figure 5 : Architecture global de notre system	32
Figure 6 : Brochage d'un PIC 16F876A	33
Figure 7 : Brochage d'un circuit DS1307	34
Figure 8 : Dimension d'un double afficheur 7 segment.....	35
Figure 9 : Schéma d'un double afficheur 7SEG-MPX2 dans Proteus	35
Figure 10 : Schéma simplifié (exemple) d'un circuit d'alimentation.....	36
Figure 11 : Sous-circuit RTC + EEPROM.....	47
Figure 12 : Sous-circuit boutons-poussoirs	48
Figure 13 : Sous-circuit date et heure.....	48
Figure 14 : Sous-circuit horaires de prière + module GPS.....	49

Sommaire

Introduction générale.....	12
Chapitre 1 : Généralités sur les systèmes temps réel et l’affichage numérique	14
1.1. Définition d’un système temps réel	14
1.2. Composants d’un système embarqué.....	14
1.3. Technologies d’affichage (LCD, 7 segments, OLED, etc.)	15
1.3.1. LCD (Liquid Crystal Display)	15
1.3.2. Affichage à sept segments.....	17
1.3.3. OLED (Organic Light Emitting Diode).....	18
1.4. Les horloges temps réel (RTC) : DS1307, DS3231, etc.....	19
1.4.1. Fonctionnement des RTC.....	20
1.4.2. Caractéristique du DS 1307	20
1.5. Les registres à décalage (ex. : 74HC595)	21
1.6. Principe de fonctionnement	21
1.7. Exemple de fonctionnement	22
Chapitre 2 : Calcul des temps de prière à partir des coordonnées géographiques	26
2.1. Principes astronomiques des horaires de prière.....	26
2.2. Paramètres nécessaires (latitude, longitude, fuseau horaire, angle solaire).....	26
2.3. Méthodes de calcul (formules, bibliothèques existantes).....	27
2.4. Présentation d’un algorithme de calcul des horaires	28
Chapitre 3 : Conception matérielle du système.....	32
3.1. Architecture globale du système.....	32
3.2. Choix du microcontrôleur (PIC, Arduino, ESP, etc.).....	33
3.3. Schéma électronique.....	34
3.3.1. Horloge temps réel (RTC).....	34
3.3.2. Afficheurs 7 segments.....	34
3.3.3. Circuit d’alimentation	36
3.4. Simulation sous Proteus.....	36
3.4.1. Préparation du projet dans Proteus.....	36
3.4.2. Simulation du système et tests	37
3.5. Réalisation de la maquette	37
3.5.1. Sélection des composants.....	37

3.5.2. Montage sur plaque de test (breadboard).....	37
3.5.3. Optimisation et corrections	37
Chapitre 4 : Programmation et affichage des horaires de prière	40
4.1. Organigramme du programme.....	40
4.2. Environnement de développement (MikroC, Arduino IDE, etc.)	41
4.3. Lecture de l'heure réelle via RTC	42
4.4. Intégration de l'algorithme de calcul des horaires.....	43
4.5. Multiplexage de l'affichage	43
4.5.1. Principe du multiplexage :	43
4.5.2. Mise en œuvre avec le circuit MAX7219	44
4.5.3. Logique de multiplexage logiciel (sans MAX7219).....	44
4.5.4. Intégration au système global :	45
4.6. Test et validation du système.....	45
Chapitre 5 : Résultats expérimentaux et perspectives	47
5.1. Présentation des résultats	47
5.1.1. L'horloges temps réel (RTC) et mémoire EEPROM.....	47
5.1.2. Les boutons de configuration	47
5.1.3. La date et ses composantes connectées	48
5.1.4. Les temps de prière	49
5.2. Limites du système	49
5.3. Améliorations possibles.....	50
5.3.1. GPS	50
5.3.2. WiFi	50
5.3.3. Affichage TFT.....	50
5.3.4. Application Mobile	50
6. Conclusion générale	52

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Résumé

Ce mémoire traite de la conception et de la mise en œuvre d'un système embarqué autonome destiné à l'affichage automatisé des horaires des cinq prières quotidiennes prescrites en islam. S'appuyant sur une architecture fondée sur un microcontrôleur PIC16F876, le système combine plusieurs modules complémentaires, notamment une horloge temps réel (RTC), un récepteur GPS, un circuit d'affichage à LED 7 segments ainsi qu'un ensemble de dispositifs de commande locale.

La première partie du travail est consacrée à l'étude des fondements théoriques et des paramètres astronomiques nécessaires au calcul des horaires de prière, à partir des coordonnées géographiques et des cycles solaires. Ces calculs ont été traduits sous forme d'algorithmes implémentés directement dans l'environnement embarqué du microcontrôleur.

Sur le plan matériel, le circuit a été conçu de manière modulaire afin de garantir la fiabilité, la facilité de maintenance et la possibilité d'évolutivité. Une attention particulière a été portée à l'optimisation énergétique ainsi qu'à la robustesse du circuit d'alimentation.

Les résultats obtenus confirment la pertinence de l'approche adoptée, tant au niveau de la précision horaire que de la stabilité fonctionnelle du système. Enfin, des perspectives d'amélioration sont évoquées, notamment l'ajout d'un affichage TFT, l'intégration du WiFi, ou encore le développement d'une application mobile permettant le contrôle et la configuration à distance.

Mots-clés : RTC ; GPS ; PIC ; I2C ; EEPROM ; PWM ; RAM

ملخص

تتناول هذه الأطروحة تصميم وتنفيذ نظام مدمج مستقل للعرض الآلي لمواقيت الصلوات الخمس المنصوص عليها في الإسلام. استناداً إلى بنية تعتمد على جهاز استقبال النظام العالمي لتحديد المواقع، (RTC)، يجمع النظام بين عدة وحدات تكميلية، بما في ذلك ساعة الوقت الحقيقي PIC16F876 متحكم دقيق ذات 7 أجزاء ومجموعة من أجهزة التحكم المحلية LED ودائرة عرض

خُصص الجزء الأول من العمل لدراسة الأسس النظرية والمعايير الفلكية اللازمة لحساب مواقيت الصلاة، بناءً على الإحداثيات الجغرافية والدورات الشمسية. وقد تُرجمت هذه الحسابات إلى خوارزميات تم تنفيذها مباشرةً في البيئة المدمجة في المتحكم الدقيق.

أما من ناحية الأجهزة، فقد تم تصميم الدائرة بطريقة معيارية لضمان الموثوقية وسهولة الصيانة وقابلية التوسع. تم إيلاء اهتمام خاص لتحسين الطاقة ومتانة دائرة إمداد الطاقة.

تؤكد النتائج التي تم الحصول عليها أهمية النهج المعتمد، سواء من حيث دقة الوقت أو الاستقرار الوظيفي للنظام. وأخيراً، ذُكر عدد من التحسينات الممكنة، ودمج الواي فاي، وتطوير تطبيق للهاتف المحمول للتحكم عن بُعد والتكوين TFT بما في ذلك إضافة شاشة

الكلمات المفتاحية: RTC ; GPS ; PIC ; I2C ; EEPROM ; PWM ; RAM

Abstract

This thesis deals with the design and implementation of an autonomous embedded system for the automated display of the times of the five daily prayers prescribed in Islam. Based on an architecture based on a PIC16F876 microcontroller, the system combines several complementary modules, including a real-time clock (RTC), a GPS receiver, a 7-segment LED display circuit and a set of local control devices.

The first part of the work is devoted to studying the theoretical foundations and astronomical parameters needed to calculate prayer times, based on geographical coordinates and solar cycles. These calculations were translated into algorithms implemented directly in the microcontroller's on-board environment.

On the hardware side, the circuit has been designed in a modular way to ensure reliability, ease of maintenance and scalability. Particular attention has been paid to energy optimisation and the robustness of the power supply circuit.

The results obtained confirm the relevance of the approach adopted, both in terms of time accuracy and the functional stability of the system. Lastly, a number of possible improvements are mentioned, including the addition of a TFT display, the integration of WiFi, and the development of a mobile application for remote control and configuration.

Keywords : RTC ; GPS ; PIC ; I2C ; EEPROM ; PWM ; RAM

Introduction générale

Introduction générale

Les systèmes en temps réel tiennent un rôle crucial dans divers domaines actuels, allant de la gestion industrielle aux applications domestiques. Ils se distinguent par leur aptitude à réaliser des tâches avec une exactitude rigoureuse, répondant ainsi aux besoins d'une société de plus en plus automatisée.

Pour illustrer, imaginez un lieu une mosquée, située dans une zone isolée où les heures de prière peuvent fluctuer en fonction des spécificités géographiques. En incorporant un dispositif intégré équipé d'un module GPS, les membres de cette mosquée ont la possibilité d'obtenir une indication automatique et précise des heures de prière, en concordance avec leur localisation géographique exacte. Ce dispositif diminue les erreurs possibles et garantit une précision cruciale pour l'observance des rituels religieux.

Le but de ce mémoire est la conception et la réalisation d'un système d'affichage numérique qui montre l'heure, le calendrier et les horaires de prière. Ce dispositif qui utilise les coordonnées géographiques récupérées grâce à un module GPS, a pour but de proposer une solution précise, fiable et indépendante répondant aux demandes spécifiques de la communauté musulmane. Il envisage aussi de prouver la capacité technique de l'intégration entre les technologies modernes (systèmes intégrés, algorithmes astronomiques et affichage numérique) afin de répondre à des défis pratiques et sociétales.

En connectant des technologies comme les systèmes embarqués, les algorithmes de calculs de prières et les technologies d'affichage, ce projet vise non seulement à répondre à un besoin particulier, mais aussi à démontrer la flexibilité et l'efficacité des solutions en temps réel dans des situations pratiques.

Chapitre 1

Généralités sur les systèmes temps réel et l'affichage numérique

Chapitre 1 : Généralités sur les systèmes temps réel et l’affichage numérique

Ce chapitre présente les bases des systèmes embarqués temps réel, ainsi que les technologies d’affichage pertinentes pour notre projet, telles que les afficheurs LCD, 7 segments et OLED.

1.1. Définition d’un système temps réel

Un système temps réel est un dispositif informatique conçu pour fonctionner en respectant des contraintes temporelles sévères, où la célérité du traitement est tout aussi cruciale que la précision des résultats. Contrairement aux systèmes informatiques traditionnels, un système en temps réel doit fournir des réponses dans des délais prédéfinis, souvent essentiels pour le fonctionnement optimal du processus qu’il supervise.

On utilise couramment ces systèmes dans des secteurs où le respect des délais est crucial, comme l’aéronautique (systèmes de pilotage intégrés), l’industrie (gestion de processus dans les usines), ou encore le domaine médical (systèmes de contrôle en temps). On fait habituellement la distinction entre deux sortes de systèmes en temps réel :

Temps réel strict (hard real-time) : où aucun dépassement des délais n’est toléré, comme dans les systèmes de contrôle de vol.

Temps réel souple (soft real-time) : où des dépassements mineurs sont acceptables, comme dans les jeux en réseau. [1] [2]

1.2. Composants d’un système embarqué

Un système embarqué est formé de divers éléments essentiels qui travaillent ensemble pour accomplir des tâches précises en temps réel. Parmi les composants, on retrouve :

- **Processeur ou Microcontrôleur** : Le cœur du système embarqué, responsable du traitement des données et de l’exécution des instructions. Il peut s’agir d’un microprocesseur (ex. : ARM Cortex) ou d’un microcontrôleur (ex. : Arduino, PIC, ESP32)
- **Mémoire** :
 - **RAM (mémoire vive)** : utilisée pour le stockage temporaire des données pendant l’exécution.
 - **ROM (mémoire morte)** : contient les instructions de base nécessaires au démarrage du système.
 - **Mémoire Flash** : utilisée pour stocker le programme et les données persistantes

- Interfaces de communication : Les systèmes embarqués utilisent des protocoles comme UART, SPI, I2C ou CAN pour échanger des données entre les composants internes ou avec des dispositifs externes.
- Modules d'alimentation : Fournissent l'énergie nécessaire au fonctionnement du système, souvent via des batteries ou des convertisseurs de tension.
- Horloge système : Un oscillateur interne qui synchronise les opérations du système et garantit le respect des contraintes temporelles.
- Capteurs : Ces dispositifs collectent des données environnementales (ex. : température, pression, lumière) et les transmettent au processeur pour analyse [3]

1.3. Technologies d'affichage (LCD, 7 segments, OLED, etc.)

Les technologies d'affichage ont une importance cruciale dans la visualisation des informations. Que l'on parle d'affichage de l'heure, du calendrier ou d'informations spécifiques, le type d'afficheur à utiliser est déterminé par les besoins spécifiques de l'application, comme la lisibilité de la lecture, la consommation d'énergie ou la complexité de l'implémentation. [4]

1.3.1. LCD (Liquid Crystal Display)

Les écrans LCD utilisent des cristaux liquides pour ajuster la lumière et afficher des images. Ils se composent de plusieurs niveaux, incluant une couche de cristaux liquides, des substrats en verre, des électrodes transparentes et un rétroéclairage. Cette technologie est couramment employée grâce à sa consommation d'énergie réduite et son prix raisonnable. [4]

1.3.1.1. Structure de base

Les écrans LCD (Liquid Crystal Display) fonctionnent grâce à une mince couche de cristaux liquides positionnée entre deux substrats en verre. Ces cristaux liquides ont la capacité de changer leur orientation lorsqu'un champ électrique est appliqué, ce qui offre un moyen de réguler la lumière qui passe à travers eux. Les cristaux sont contenus entre deux plaques de verre claire, chacune dotée d'électrodes transparentes qui sont souvent composées d'oxyde d'indium-étain (ITO). Deux polariseurs, disposés de chaque côté du système, sont utilisés pour faire une sélection de la lumière. [4]

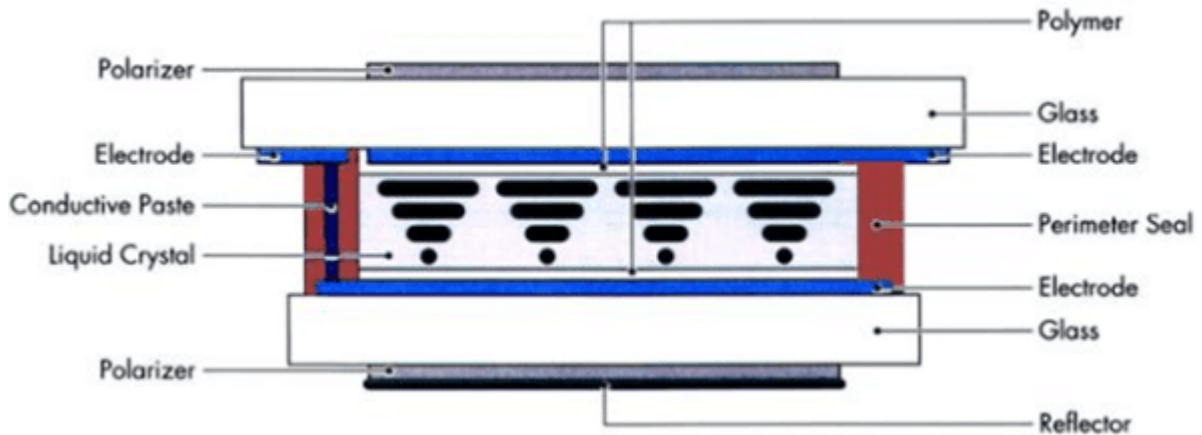


Figure 1 : Structure de l'écran LCD [4]

Rétroéclairage

À l'inverse de technologies telles que l'OLED, les écrans LCD ne génèrent pas de lumière par eux-mêmes. Ces derniers nécessitent une source lumineuse externe, généralement un rétroéclairage LED ou une lampe fluorescente à cathode froide (CCFL). Ce rétroéclairage génère une lumière qui passe d'abord par le premier polariseur avant d'interagir aux cristaux liquides. [4]

Orientation des cristaux liquides

En l'absence de champ électrique, les cristaux liquides sont orientés de façon à déformer la lumière polarisée, ce qui la permet de traverser le second polariseur. Quand les cristaux liquides sont exposés à un champ électrique, ils se positionnent de manière à bloquer la lumière, ce qui rend le pixel correspondant sombre. [5]

Formation des couleurs

Dans le cas des écrans en couleur, chaque pixel est divisé en sous-pixels de couleurs rouge, vert et bleu (RVB). En ajustant la luminosité qui passe à travers chacun de ces sous-pixels, on peut générer des millions de couleurs différents. Il existe diverses sortes d'écrans LCD, comme les modèles TN (Twisted Nematic) qui se distinguent par leur rapidité de réponse, ou les modèles IPS (In-Plane Switching), qui sont appréciés pour leur large angle de vision et leur supériorité en termes de reproduction des couleurs. [4]

1.3.1.2. Applications

On utilise souvent cette technologie dans des appareils comme les montres électroniques, les calculatrices, les écrans d'ordinateurs portables et les téléviseurs, en raison de sa faible consommation énergétique, de son prix abordable et de sa clarté optimale dans diverses conditions.

1.3.2. Affichage à sept segments

Les afficheurs à sept segments sont des dispositifs d'affichage numériques simples et largement utilisés. Leur structure, qui se compose de sept segments lumineux formant un « 8 », offre une représentation claire et facile à lire. Ces afficheurs, par leur facilité d'utilisation et leur performance, sont régulièrement incorporés dans des dispositifs pratiques tels que les montres digitales, les calculatrices et les équipements domestiques.

1.3.2.1. Structure et Fonctionnement

Les afficheurs à 7 segments sont des systèmes d'affichage numériques basiques, constitués de sept segments lumineux disposés en forme de chiffre « 8 ». Chaque segment est une diode électroluminescente (LED) ou une cellule à cristaux liquides (LCD), et porte une lettre d'identification allant de A à G. Il est possible d'activer ou de désactiver ces segments un par un afin de créer des chiffres ou des caractères spécifiques. Par exemple, pour représenter le numéro « 7 », les segments A, B et C s'activent, alors que les segments D, E, F et G restent éteints. [6]

Chaque segment est relié à une broche de commande, ce qui donne la possibilité de gérer son état (activé ou désactivé) par le biais d'un circuit électronique. Les afficheurs à 7 segments peuvent être configurés en mode anode commune ou cathode commune. Dans une configuration à anode commune, chaque anode des segments est reliée entre elles et reçoit une alimentation en tension positive, tandis que les cathodes sont gérées séparément. En revanche, dans une configuration à cathode commune, toutes les cathodes sont connectées entre elles tandis que les anodes sont contrôlées d'une manière individuelle. [7]

Ces afficheurs peuvent aussi comporter un point décimal (DP) pour représenter avec exactitude les valeurs numériques, comme dans les horloges ou les calculatrices. Ce point est généralement illustré par une LED ou un autre élément, qui se trouve à côté des segments principaux. [6]

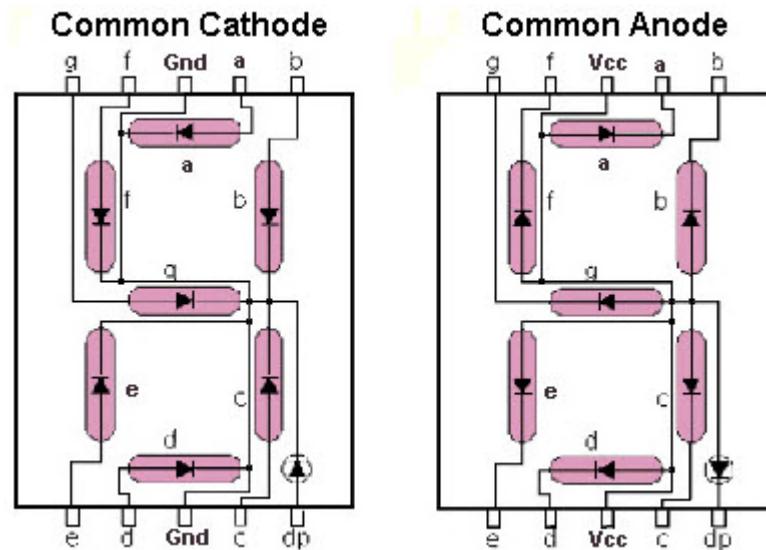


Figure 2 : Configuration d'un afficheur 7 segments [26]

Les afficheurs 7 segments sont fréquemment utilisés avec des microcontrôleurs ou des circuits intégrés dédiés, tels que les décodeurs BCD (Binary Coded Decimal), afin de faciliter leur gestion. Ces décodeurs transforment un signal numérique en un schéma d'activation des segments qui correspond au chiffre à présenter.

1.3.3. OLED (Organic Light Emitting Diode)

Les écrans OLED fonctionnent grâce à des matières organiques qui produisent de la lumière lorsqu'ils sont soumis à un courant électrique. Contrairement aux LCD, ils ne nécessitent pas de rétroéclairage, ce qui permet d'obtenir un niveau de noir significative et un contraste élevé.

1.3.3.1. Structure et Fonctionnement

La structure d'un pixel OLED comprend plusieurs éléments clés. La couche de substrat, qui est généralement en verre ou en plastique, constitue la base pour les autres couches. Les électrodes, appelées anode et de cathode, permettent le passage du courant électrique à travers les substances organiques. La génération de lumière est l'accomplissement de la couche émissive, alors que les couches qui assurent le transport des charges dirigent les électrons et les trous (lacunes d'électrons) vers la couche émissive afin d'optimiser l'efficacité lumineuse. Quand le courant passe à travers ces couches, il stimule les molécules organiques, générant par conséquent une émission de photons, c'est-à-dire de lumière visible. [8] [9]

Une autre caractéristique remarquable des écrans OLED est leur aptitude à être produits sur des substrats flexibles. Cela permet la fabrication d'écrans courbés, pliables ou même déroulables, ouvrant ainsi la porte à des conceptions innovantes pour les appareils électroniques. De plus, grâce à leurs sous-pixels rouge, vert et bleu qui s'illuminent d'une manière indépendante pour produire des images en couleur, les écrans OLED présentent une vaste palette de couleurs. En ajustant l'intensité du courant électrique dans chaque sous-pixel, on peut reproduire des millions de variations de couleurs. [9]

1.3.3.2. Avantages

Les écrans OLED présentent plusieurs avantages notables en comparaison avec d'autres technologies d'affichage. En raison de leur capacité à produire leur propre lumière, ils offrent des couleurs brillantes et précises, ainsi qu'un fort contraste. Avec des noirs profonds et des blancs lumineux, il est possible d'identifier les plus subtils détails dans les images. Par ailleurs, les écrans OLED offrent des temps de réponse rapides et un large angle de vision, ce qui optimise l'expérience de l'utilisateur.

1.3.3.3. Applications

La technologie OLED est utilisée dans divers équipements électroniques, comme les smartphones, les téléviseurs, les montres intelligentes et les panneaux d'affichage. Elle est très valorisée dans les produits haut de gamme grâce à sa qualité d'image exceptionnelle et sa possibilité d'être produite sur des substrats flexibles, ce qui permet la réalisation d'écrans courbés ou pliables. [10]

1.3.3.4. Limitations

Bien que la technologie OLED offre de nombreux avantages, elle n'est pas sans limitations. La longévité des écrans OLED est limitée en raison de la détérioration des matériaux organiques. Ils sont aussi sensibles à l'eau et à l'oxygène, ce qui exige des procédures de production spécifiques pour les préserver. Pour finir, le coût de production plus élevé de ceux-ci par rapport aux écrans LCD, ce qui peut limiter leur intégration dans les produits économiques. [10]

1.4. Les horloges temps réel (RTC) : DS1307, DS3231, etc.

Les horloges temps réel (RTC, Real-Time Clock) ont été développés pour garantir une mesure exacte du temps, même en présence d'une interruption de l'alimentation. Ces dispositifs sont couramment employés dans les systèmes intégrés pour assurer la continuité de l'heure et de la date. Leur opération est basée sur des oscillateurs internes produisant une fréquence stable. Ces composants sont

Chapitre 1 : Généralités sur les systèmes temps réel et l’affichage numérique

indispensables dans des applications comme les systèmes de contrôle, les dispositifs d'enregistrement et les équipements automatisés. Dans notre circuit, nous utiliserons le DS1307. [11]

1.4.1. Fonctionnement des RTC

Les horloges en temps réel s'utilisent un oscillateur à quartz, généralement réglé sur une fréquence de 32,768 kHz, comparable à celui utilisé dans les montres à quartz. Cette fréquence est optimale pour les circuits numériques, étant équivalente à 2^{15} cycles par seconde, ce qui simplifie le décompte des secondes, minutes, heures et plus. Les circuits de (RTC) sont fréquemment incorporés dans des circuits intégrés comportant des registres pour conserver l'heure et la date actuelles. Ces registres peuvent être configurés ou lus par le biais d'interfaces de communication telles qu'I2C ou SPI. [12]

1.4.2. Caractéristique du DS 1307

1.4.2.1. Caractéristiques de mesure du temps

Le DS1307 garde en mémoire les secondes, minutes, heures, jours, mois et années, tout en tenant compte des années bissextiles. Il est capable de fonctionner en format 12 heures (avec indication AM/PM) ou 24 heures. Cette flexibilité le rend approprié pour une multitude d'applications. [13]

1.4.2.2. Interface de communication I2C

Le module utilise une interface série I2C bidirectionnelle, ce qui rend son liaison à des microcontrôleurs tels qu'Arduino plus facile. Cette interface diminue le nombre de broches requises pour la communication, rendant ainsi le câblage plus simple. [13]

1.4.2.3. Mémoire non volatile intégrée

Le DS1307 possède 56 octets de mémoire RAM non volatile (NVRAM), ce qui permet de conserver des informations utilisateur. Une batterie de secours conserve cette mémoire. [13]

1.4.2.4. Batterie de secours

Une des caractéristiques clé du DS1307 est sa capacité de fonctionner avec une batterie de secours (par exemple CR2032). [13]

1.4.2.5. Sortie d'onde carrée programmable

Le circuit est équipé d'une sortie qui produit un signal carré. [13]

1.4.2.6. Consommation énergétique faible

En fonctionnement sur batterie, le DS1307 utilise moins de 500 nA, ce qui prolonge considérablement la longévité de la pile. [13]

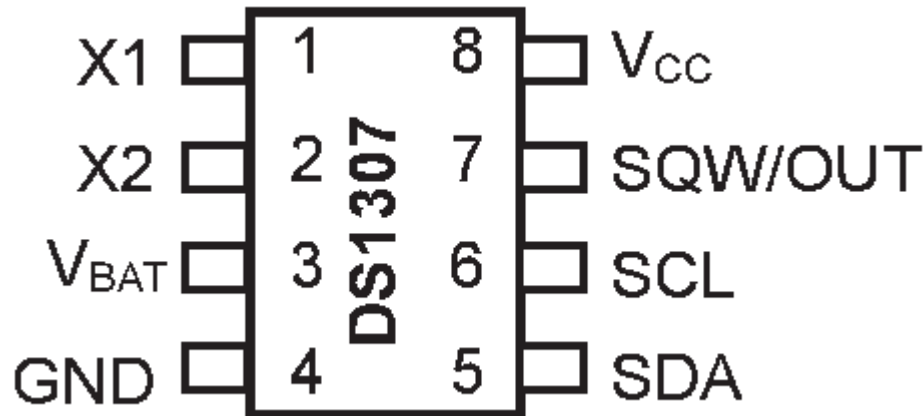


Figure 3 : Brochage d’un circuit DS1307 [27]

1.5. Les registres à décalage (ex. : 74HC595)

Dans les systèmes numériques, une gestion efficace des signaux et des données est cruciale, spécialement lorsque le nombre de sorties disponibles sur un microcontrôleur est limité. Les registres à décalage facilitent cette gestion en proposant une technique de stockage et de transmission des bits via une ligne série, avant leur distribution en format parallèle. [14]

1.6. Principe de fonctionnement

Les registres à décalage sont des composants électroniques conçus pour conserver et transmettre des informations binaires d’une façon séquentielle. Ils sont composés d’une suite de bascules synchrones reliés en série, facilitant le transfert des bits d’information à chaque impulsion d’horloge. Ce processus est crucial dans les systèmes numériques pour la transformation des données d’un format série à un format parallèle. [14]

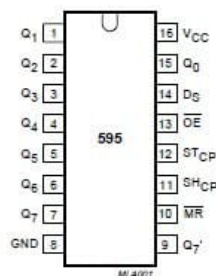
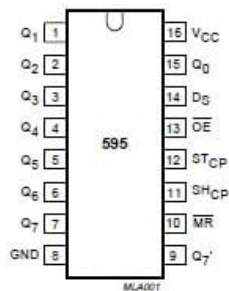


Figure 4 : Brochage d’un circuit 74HC595 [28]

Le fonctionnement d'un registre à décalage est basé sur l'injection d'un signal d'horloge qui synchronise le passage des bits d'une bascule à l'autre. Quand une impulsion d'horloge est donnée, l'état logique de chaque bascule se transmet à la suivante, provoquant ainsi un effet de décalage. Par exemple, dans un registre à décalage de 4 bits, un bit qui se trouve dans la bascule initiale sera transféré à la deuxième bascule lors de la première impulsion. Ce transfert continuera vers la troisième lors de l'impulsion suivante, et ainsi de suite. [14]

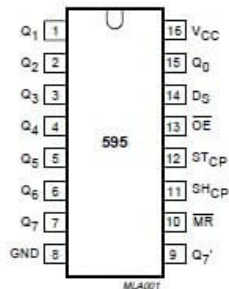
1.7. Exemple de fonctionnement

Nous allons écrire la valeur "10010001" sur la sortie, nous commençons par injecter les valeurs dans DS puis injectons une impulsion dans SH_CP



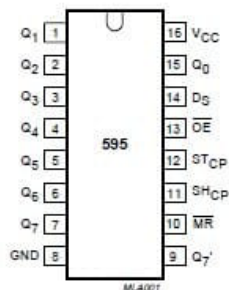
Ds=1	0	0	0	0	0	0	0	0
SH_CP=0								

Qn	0	0	0	0	0	0	0	0
----	---	---	---	---	---	---	---	---



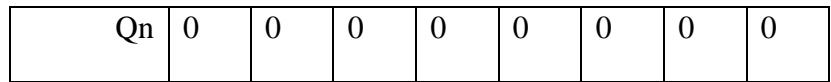
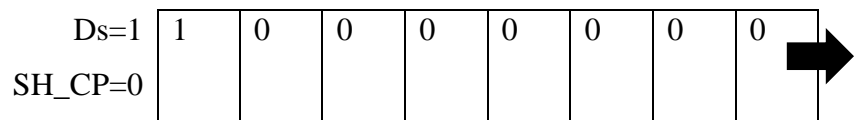
Ds=1	0	0	0	0	0	0	0	0
SH_CP=1								

Qn	0	0	0	0	0	0	0	0
----	---	---	---	---	---	---	---	---

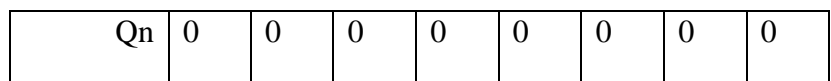
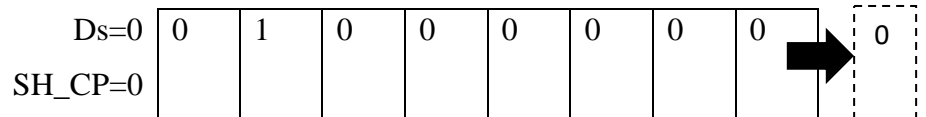
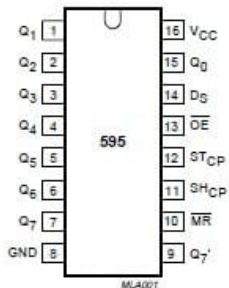
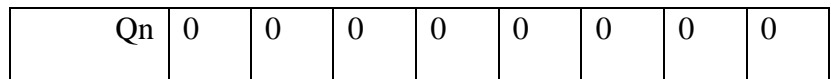
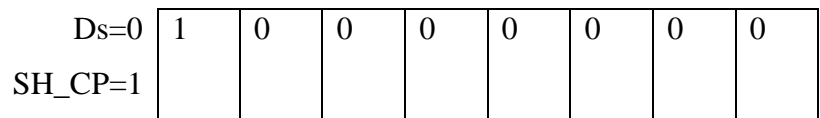
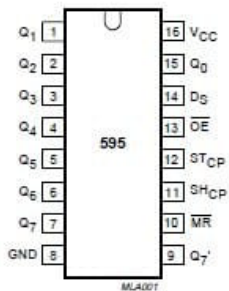
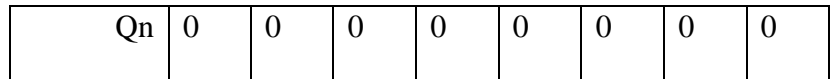
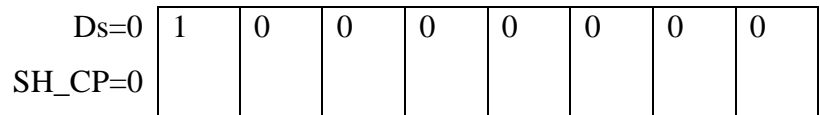
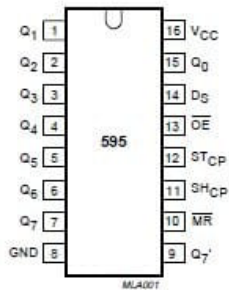


0

Chapitre 1 : Généralités sur les systèmes temps réel et l'affichage numérique



Nous avons maintenant réussi à injecter le nombre « 1 » dans la mémoire interne, passons au suivant



Chapitre 1 : Généralités sur les systèmes temps réel et l'affichage numérique

Nous avons maintenant réussi à injecter le nombre « 0 » dans la mémoire interne, nous répéterons les étapes jusqu'à ce que nous ayons le numéro « 10010001 » dans la mémoire interne, après cela nous injectons un signal (0 → 1 → 0) dans ST_CP (broche de stockage) qui copie la mémoire interne sur les broches de sortie Qn

Qn	1	0	0	1	0	0	0	1
----	---	---	---	---	---	---	---	---

Chapitre 2 :

Calcul des temps de prière à partir des coordonnées géographiques

Chapitre 2 : Calcul des temps de prière à partir des coordonnées géographiques

Ce chapitre décrit les fondements astronomiques nécessaires au calcul des horaires de prière en fonction de la position géographique, et les méthodes employées pour les obtenir par algorithmes ou bibliothèques.

2.1. Principes astronomiques des horaires de prière

En islam, les temps de prière sont essentiellement fixés en fonction de la position visible du Soleil dans le ciel. Ces horaires sont basés sur des principes astronomiques précis, qui peuvent légèrement différer en fonction des écoles de pensée et des méthodes de calcul, mais les bases demeurent généralement identiques. Voici un résumé des principes astronomiques liés à chaque prière :

- **FAJR (aube) :** Quand la lumière du jour commence à se manifester horizontalement à l'est, le Soleil se trouve approximativement à -18° sous l'horizon (cette valeur est couramment utilisée, mais elle peut varier entre -15° et -18° en fonction des techniques employées) et se termine juste avant l'apparition du Soleil.
- **DHUHR (midi) :** Peu après que le Soleil ait atteint son apogée (culmination) et commence à descendre vers l'ouest, il dépasse son point culminant (solaire). Le moment précis du zénith fluctue selon la latitude et la saison, se terminant lorsque l'ombre d'un objet est équivalente à sa taille (selon la méthode hanafite, l'ombre doit être deux fois plus longue que l'objet lui-même).
- **ASR (après-midi) :** Lorsque l'ombre d'un objet atteint une proportion 1x sa hauteur selon l'école majoritaire et 2x selon l'école hanafite et finir juste avant le coucher du Soleil.
- **MAGHRIB (coucher du soleil) :** Immédiatement après le coucher du Soleil (hauteur du Soleil = 0°) et finir lorsqu'il reste un peu de lumière crépusculaire (début d'ISHA).
- **ISHA (nuit) :** Lorsque la lumière rouge du crépuscule (Shafaq Ahmar) disparaît entièrement du ciel. L'heure à laquelle le soleil descend sous l'horizon est approximativement entre -17° et -18° , il termine avant le FAJR (aube), mais il est recommandé de la prier avant minuit solaire.

2.2. Paramètres nécessaires (latitude, longitude, fuseau horaire, angle solaire)

Pour déterminer les heures de prière, il est nécessaire d'avoir les paramètres suivants :

Chapitre 2 : Calcul des temps de prière à partir des coordonnées géographiques

- **Latitude** : La position nord ou sud par rapport à l'équateur (en degrés) qui détermine la trajectoire apparente du Soleil dans le ciel à un endroit donné et Influence la durée du jour et la hauteur du Soleil à différents moments. Ex. 34.67802840° N pour Tlemcen.
- **Longitude** : Cette localisation est définie par rapport à l'est ou à l'ouest du méridien de Greenwich (en degrés), elle permet de déterminer l'heure locale moyenne et d'adapter les horaires conformément à la position relative au fuseau horaire central et permet d'ajuster les horaires solaires au temps standard local. Ex. -1.36621600° E pour Tlemcen.
- **Fuseau horaire** : Le décalage horaire par rapport à l'heure GMT/UTC qui convertit les horaires calculés en temps solaire en heure légale locale et il est important pour afficher les horaires de prière selon l'heure officielle du pays. Ex. UTC+1 pour Tlemcen.
- **L'angle solaire** : Angle entre le Soleil et l'horizon, utilisé surtout pour FAJR et ISHA et Permet de définir le moment où la lumière du Soleil commence à apparaître ou disparaître.
 - Angle du Soleil pour FAJR : généralement entre -15° et -18°
 - Angle du Soleil pour ISHA : généralement entre -15° et -18°
 - Méthode de calcul d'ASR :
 - Standard (Shafi'i, Maliki et Hanbali) : Ombre = 1x objet
 - Hanafi : Ombre = 2x objet

2.3. Méthodes de calcul (formules, bibliothèques existantes)

Le calcul des horaires de prière se fonde sur des formules astronomiques exactes liées à la position du soleil, ainsi que sur des bibliothèques logicielles qui facilitent leur mise en œuvre. Voici un résumé des principales techniques de calcul, incluant les formules et les outils disponibles.

- Formules de base utilisées :

Les horaires sont calculés à partir de l'angle horaire du Soleil pour chaque prière, dérivé de la trigonométrie sphérique.

- Formule générale de l'angle horaire :

$$\cos(H) = \frac{\sin(A) - \sin(\varphi) * \sin(\delta)}{\cos(\varphi) * \cos(\delta)}$$

- H : angle horaire (en degrés)
- A : angle solaire spécifique pour la prière (ex. -18° pour FAJR)
- φ : latitude de l'observateur

Chapitre 2 : Calcul des temps de prière à partir des coordonnées géographiques

- δ : déclinaison solaire (varie chaque jour) [31]
- Temps solaire pour un angle donné :
$$T = 12 \pm \frac{H}{15}$$
 - Conversion de l'angle horaire en heure solaire locale.
 - Le signe \pm dépend de la prière (matin ou soir).
- Correction pour l'équation du temps et le fuseau horaire :
$$T_{\text{local}} = T + \text{équation du temps} + \text{Décalage de longitude} + \text{Fuseau horaire}$$
 - L'équation du temps corrige l'irrégularité du mouvement solaire apparent.
 - La correction de longitude ajuste l'heure solaire au fuseau officiel. [31]
- Bibliothèques et outils de calcul existants :
 - PrayTimes.js : Un projet islamique visant à fournir une bibliothèque open source pour calculer les heures de prière musulmanes et supporter plusieurs méthodes : ISNA, MWL, Umm al-Qura, etc.
 - PyPrayTimes : Utilise les mêmes formules pour les angles solaires et les fuseaux horaires et facile à intégrer dans des projets web/app.
 - Aladhan API : API en ligne gratuite pour récupérer les horaires de prière requiert latitude, longitude, méthode et fuseau horaire.
 - Autres bibliothèques :
 - Adhan (by Batoul Apps)
 - IslamicFinder libraries

2.4. Présentation d'un algorithme de calcul des horaires

Voici une présentation claire et structurée d'un algorithme de calcul des horaires de prière, basé sur les principes astronomiques et les paramètres déjà évoqués :

- Étape 1 : Entrées nécessaires
 - Latitude (φ)
 - Longitude (λ)
 - Fuseau horaire (TZ)
 - Date

Chapitre 2 : Calcul des temps de prière à partir des coordonnées géographiques

- Méthode de calcul choisie (ISNA, MWL, Umm al-Qura, etc.)
 - Détermine les angles solaires pour Fajr et Isha [31]
- **Étape 2 : Calculs astronomiques de base**
- Calcul du jour julien (Julian Day)
 - Permet de déterminer la position du Soleil avec précision.
 - Calcul de la déclinaison solaire (δ)
 - Dépend de la date : position apparente du Soleil par rapport à l'équateur.
 - Calcul de l'équation du temps (EoT)
 - Corrige l'écart entre temps solaire apparent et temps solaire moyen.
 - Calcul du temps solaire local (TSL)
 - Corrige l'heure solaire en fonction de la longitude et du fuseau horaire [31]
- **Étape 3 : Calculs des horaires de prière [31]**

Prière	Critère astronomique	Méthode de calcul
Fajr	Soleil à -18° (ex.)	Angle horaire H pour $A = -18^\circ$, puis $T = 12 - H/15$
Dhuhr	Soleil au zénith	Midi solaire = $12 + \text{EoT} + \text{correction longitude}$
Asr	Ombre = 1x ou 2x la hauteur objet	Calcul trigonométrique basé sur angle du Soleil
Maghrib	Coucher du Soleil	Quand altitude du Soleil = 0°
Isha	Soleil à -17° (ex.)	Angle horaire H pour $A = -17^\circ$, puis $T = 12 + H/15$

- **Étape 4 : Conversion à l'heure locale**
- Appliquer : $T_{\text{local}} = T_{\text{solaire}} + \text{fuseau horaire} + \text{correction longitude}$ [31]
- **Étape 5 : Ajustements facultatifs**
- Arrondi à la minute la plus proche.
 - Ajout d'un temps de sécurité (ex. 1–2 min de marge pour Maghrib ou FAJR).
 - Méthode Umm al-Qura : valeurs fixes (ex. 90 min après Maghrib pour ISHA). [31]
- **Exemple de pseudo-code simplifié**

Python

```
def calcul_horaires_priere(latitude, longitude, timezone, date, methode):  
    JD = calcul_jour_julien(date)  
    declinaison = calc_declinaison_solaire(JD)  
    EoT = calc_equation_du_temps(JD)  
  
    midi_solaire = 12 + timezone - (longitude / 15) - EoT  
  
    fajr = calcul_angle_heure(-methode.angle_fajr, declinaison, latitude)  
    icha = calcul_angle_heure(-methode.angle_icha, declinaison, latitude)  
    maghrib = calcul_angle_heure(0, declinaison, latitude)  
    asr = calcul_asr_angle(declinaison, latitude, methode.asr_type)  
  
    return {  
        "Fajr": midi_solaire - fajr,  
        "Dhouhr": midi_solaire,  
        "Asr": midi_solaire + asr,  
        "Maghrib": midi_solaire + maghrib,  
        "Icha": midi_solaire + icha  
    }
```

Chapitre 3

Conception matérielle du système

Chapitre 3 : Conception matérielle du système

Ce chapitre décrit les composants matériels utilisés dans le système, leur interconnexion, les choix techniques faits, ainsi que la simulation et la réalisation de la maquette.

3.1. Architecture globale du système

L'architecture globale du système définit la disposition et la connexion des éléments électroniques indispensables à l'opération d'un système embarqué. Elle repose sur divers éléments cruciaux, comme le microcontrôleur, qui assure la gestion des processus et la synchronisation des tâches, les registres à décalage qui augmentent le nombre de sorties possibles, et les écrans numériques, qui permettent l'affichage des informations.

L'intégration de ces composants sur un circuit imprimé (PCB) est réalisée à l'aide d'une conception optimisée visant à assurer fiabilité, efficacité et faible consommation d'énergie. Des interfaces de communication telles qu'I2C et SPI facilitent le partage d'informations entre les éléments, alors que le système d'alimentation assure une distribution stable de l'énergie.

Pour notre projet nous avons choisi cette architecture que nous avons passé du temps à optimiser et à adapter pour répondre à nos besoins

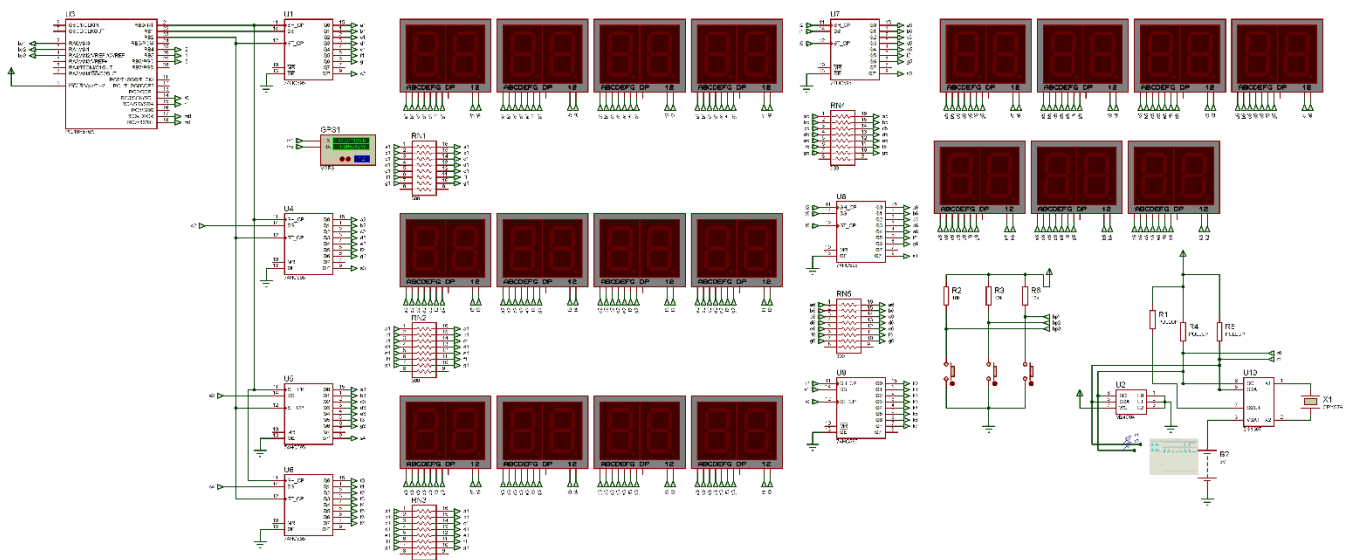


Figure 5 : Architecture globale de notre system

Chapitre 3 : Conception matérielle du système

Le microcontrôleur gère toutes les opérations de calcul et d'affichage, il lit la date et l'heure actuelles à partir du module GPS/RTC, puis les traite à l'aide d'algorithmes et de code intégré pour convertir les données d'entrée en broches de sortie qui sont ensuite transmises au registre à décalage 74HC595 en utilisant un modèle de sortie spécifique qui à son tour est poussé vers les écrans pour que l'utilisateur puisse voir le résultat final. Le microcontrôleur est également équipé et préprogrammé pour accepter l'édition manuelle de la date et de l'heure via 3 boutons poussoirs qui sont connectés à ses broches d'entrée.

3.2. Choix du microcontrôleur (PIC, Arduino, ESP, etc.)

Le choix du microcontrôleur est une phase déterminante lors de la conception d'un système intégré. Cela impacte directement les performances, l'efficacité énergétique, la simplicité de programmation et la connectivité de l'appareil. Il existe diverses familles de microcontrôleurs, chacune proposant des fonctionnalités adaptées à des exigences particulières. Il existe plusieurs types de microcontrôleurs tels que PIC, Arduino, ESP...etc

Pour notre projet, nous avons choisi le microcontrôleur PIC spécifiquement le PIC16F876A en raison de leur flexibilité et de leurs performances. Voici leurs principaux avantages : [15]

- Simplicité et accessibilité
- Large gamme de modèles
- Faible consommation énergétique
- Outils de développement performants
- Interfaces de communication variées
- Robustesse et fiabilité

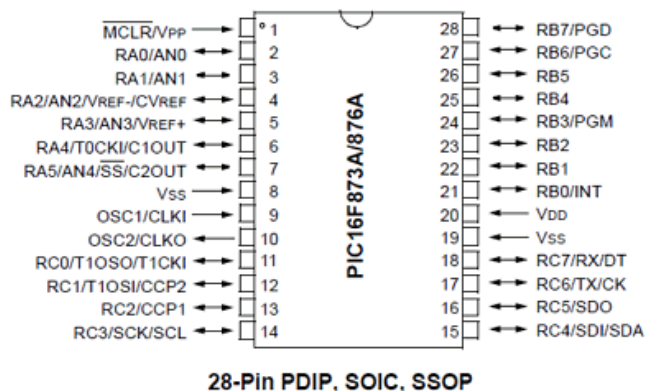


Figure 6 : Brochage d'un PIC 16F876A [29]

3.3. Schéma électronique

3.3.1. Horloge temps réel (RTC)

Le module RTC, tel que le DS1307, est connecté au microcontrôleur via l'interface I2C pour conserver l'heure et la date, même en cas de coupure d'alimentation. Il est alimenté par une pile de secours afin de garantir un fonctionnement continu.

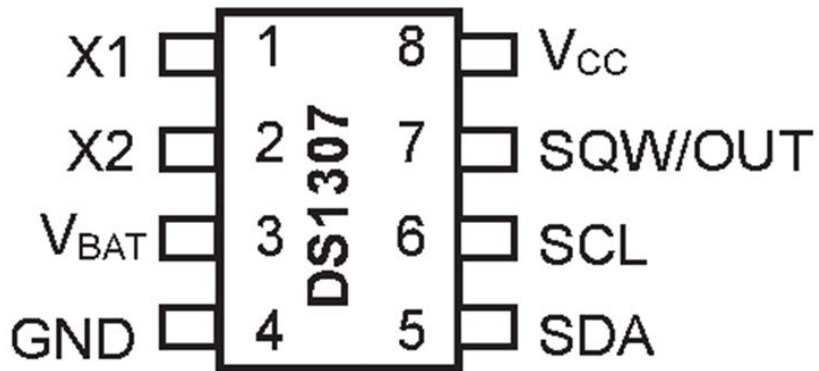


Figure 7 : Brochage d'un circuit DS1307 [27]

3.3.2. Afficheurs 7 segments

L'afficheur 7 segments est connecté au microcontrôleur via des registres à décalage qui fonctionnent selon un modèle spécifique, l'afficheur que nous avons utilisé est un double afficheur qui utilise 2 broches de sélection et 8 broches de données

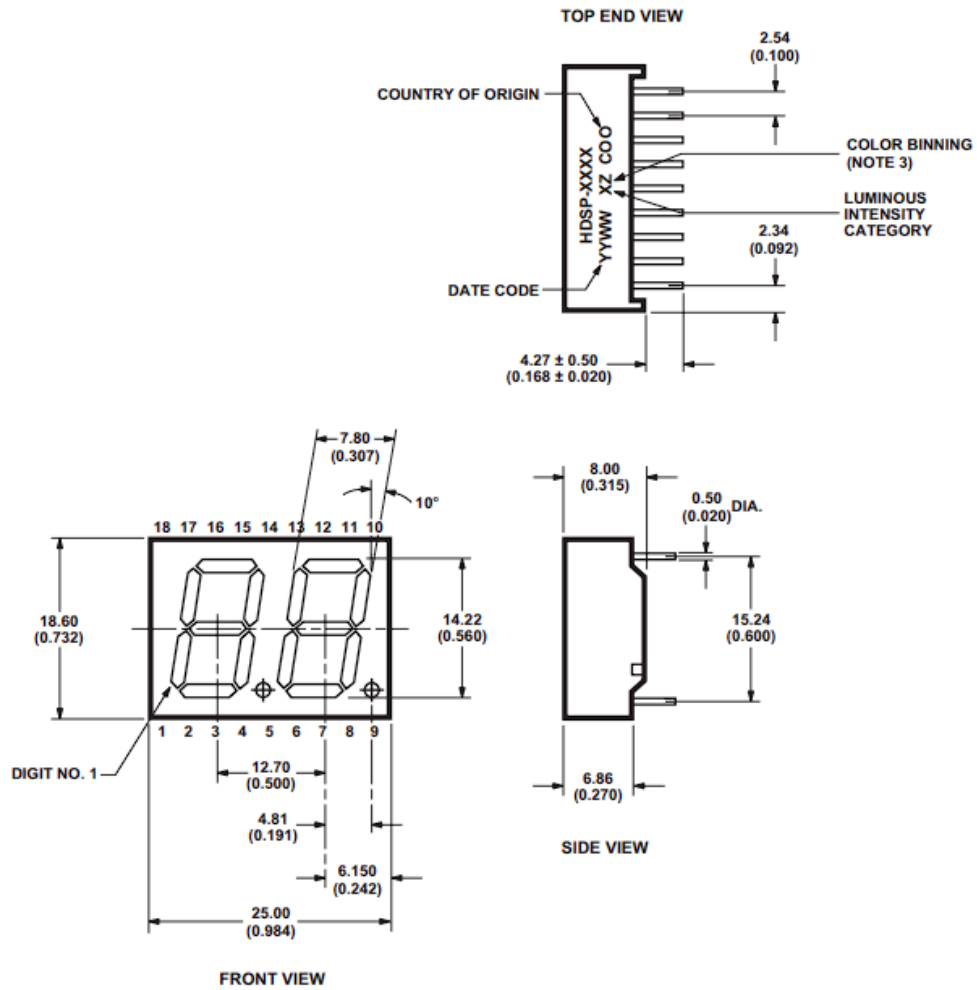


Figure 8 : Dimension d'un double afficheur 7 segment [30]

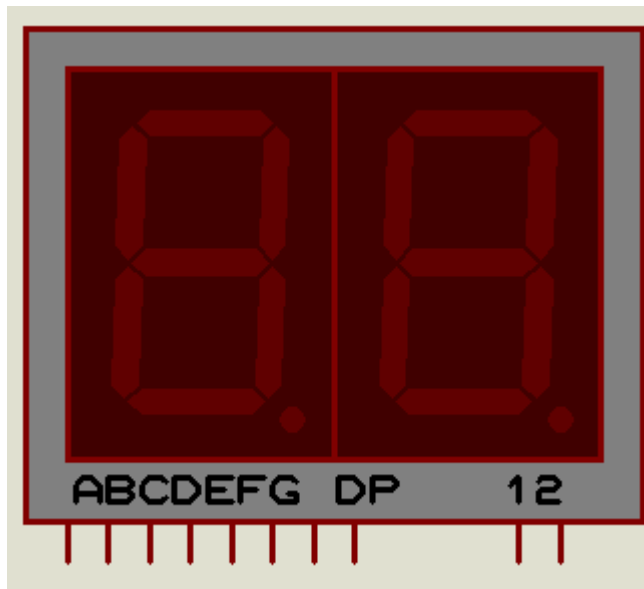


Figure 9 : Schéma d'un double afficheur 7SEG-MPX2 dans Proteus

3.3.3. Circuit d'alimentation

Le circuit d'alimentation joue un rôle crucial dans la stabilité et la fiabilité du système. Dans notre projet, il permet d'alimenter le microcontrôleur PIC16F876 ainsi que les modules. Il fournit une tension régulée de 5V. Des protections comme des diode des condensateurs de filtrage assurent la sécurité et la stabilité électrique de l'ensemble du dispositif.



Figure 10 : Schéma simplifié (exemple) d'un circuit d'alimentation

3.4. Simulation sous Proteus

La simulation d'un système électronique avant son implémentation concrète est une phase essentielle pour assurer son efficacité et identifier tout problème potentiel. Proteus est une des applications les plus couramment utilisées pour la modélisation de circuits électroniques, qui offre la possibilité d'évaluer le fonctionnement de composants comme les microcontrôleurs, afficheurs, RTC, capteurs et circuits d'alimentation.

3.4.1. Préparation du projet dans Proteus

Avant de commencer la simulation, il est essentiel de :

- Créer un nouveau projet et ouvrir la capture schématique.
- Ajouter les composants clés, tels que le microcontrôleur PIC (PIC16F876A), l'afficheur 7 segments (7SEG-MPX2), le RTC (DS1307), le registre à décalage (74HC595), et les éléments de gestion de l'alimentation.
- Relier les composants avec les bus de pour assurer l'échange des données et électricité
- Le microcontrôleur doit être configuré avec son fichier hex, généré après la compilation du code (ex. : depuis mikroC PRO for PIC). Ensuite, les paramètres de fréquence d'horloge doivent être réglés pour correspondre au circuit réel.

3.4.2. Simulation du système et tests

Une fois les connexions et configurations effectuées, la simulation est lancée pour observer le comportement des différents éléments :

- L'affichage des données sur les 7 segments
- L'évolution de l'heure du RTC et sa synchronisation avec le microcontrôleur.
- Indications de tension en temps réel (rouge, bleu)

Si des erreurs ou dysfonctionnements apparaissent, Proteus permet d'analyser les signaux et de corriger le circuit avant la phase de prototypage réel.

3.5. Réalisation de la maquette

La réalisation du prototype est une phase cruciale dans le développement du système, facilitant l'essai pratique des composants et l'évaluation de leur performance avant la production finale. Cette étape consiste à assembler physiquement les composants électroniques, à les connecter sur une plaque de prototypage (breadboard) ou une carte imprimée (PCB), et à contrôler leur compatibilité.

3.5.1. Sélection des composants

La maquette doit intégrer tous les éléments essentiels du système, notamment :

- Microcontrôleur PIC16F876
- Afficheur 7 segments
- Horloge temps réel RTC DS1307
- Registres à décalage 74HC595
- Alimentation avec régulateur de tension et protection contre les surtensions

3.5.2. Montage sur plaque de test (breadboard)

Le montage initial est généralement effectué sur une maquette, permettant un assemblage rapide et des modifications sans soudure. Cela permet de rectifier les erreurs de connexion avant la réalisation d'un circuit final.

3.5.3. Optimisation et corrections

Suite aux essais, il peut être nécessaire de faire des modifications pour optimiser les performances, rectifier d'éventuelles erreurs et perfectionner la conception avant l'étape de production finale.

Chapitre 3 : Conception matérielle du système

La création du prototype permet donc de valider l'architecture du système et d'assurer sa fiabilité avant son établissement final.

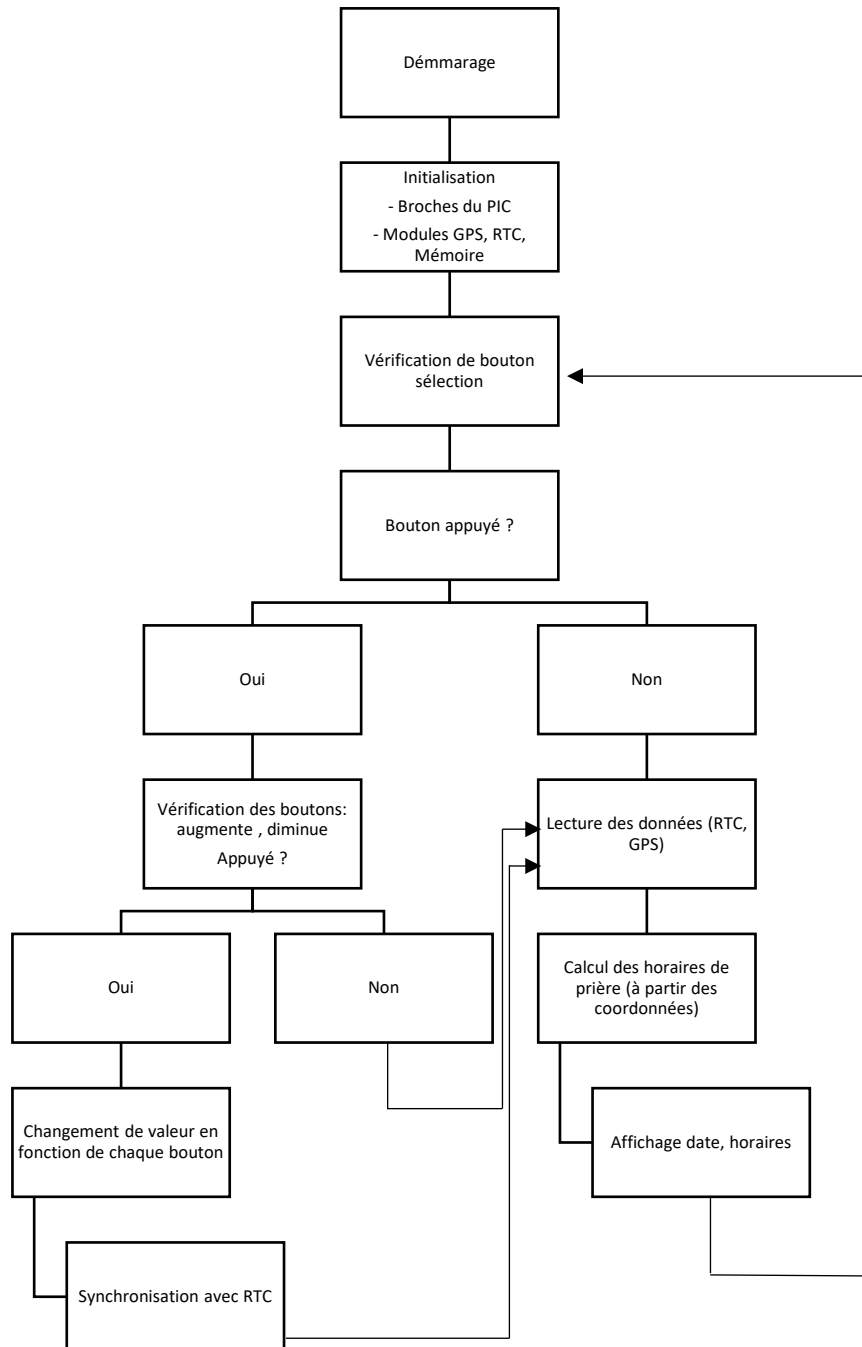
Chapitre 4 :

Programmation et affichage des horaires de prière

Chapitre 4 : Programmation et affichage des horaires de prière

Nous abordons ici la partie logicielle du système : lecture de l'heure, calcul algorithmique des prières, gestion de l'affichage et multiplexage via microcontrôleur.

4.1. Organigramme du programme



4.2. Environnement de développement (MikroC, Arduino IDE, etc.)

Dans cette partie, nous exposons les instruments logiciels employés pour la conception du système. La sélection de l'environnement de programmation est fonction de la plateforme matérielle choisie.

Dans le cadre de ce projet, nous avons utilisés :

- **Arduino IDE** : L'Arduino Integrated Development Environment (IDE) est un environnement open-source qui repose sur le langage C/C++. Bien qu'il soit principalement destiné aux cartes Arduino, ce dernier peut également être employé avec d'autres plateformes compatibles grâce à l'utilisation de bootloaders ou de bibliothèques externes. Plusieurs éléments ont guidé cette décision :
 - **Accessibilité** : Interface simple et intuitive, conçue pour convenir aux débutants qu'aux développeurs expérimentés.
 - **Communauté active** : Une vaste communauté qui propose des bibliothèques, exemples de code, forums, et assistance.
 - **Compatibilité matérielle** : Il existe de nombreuses bibliothèques, officielles et non, pour une variété étendue de capteurs, modules RTC, afficheurs LED, et circuits intégrés (MAX7219, DS3231, etc.).
 - **Portabilité** : L'IDE est compatible avec Windows, macOS et Linux, et permet de compiler et téléverser rapidement du code vers les cartes microcontrôleurs via une interface USB [16].

- **MikroC** est un environnement de développement intégré (IDE) proposé par MikroElektronika, souvent utilisé avec les microcontrôleurs PIC pour ses outils de débogage et son simulateur intégré. Il comprend :
 - Un éditeur de code C avec coloration syntaxique et suggestions de code.
 - Un compilateur optimisé générant du code binaire efficace et rapide.
 - Un simulateur intégré pour examiner le fonctionnement du programme sans nécessiter d'équipement matériel.
 - Un gestionnaire de bibliothèques permettant d'intégrer facilement des modules (LCD, RTC, I2C, SPI, PWM, etc.) [17].

Ce logiciel a été particulièrement utile pour les parties du projet nécessitant une meilleure gestion des ressources matérielles, notamment pour les cartes basées sur les microcontrôleurs PIC16F877A ou PIC18F4550, largement utilisées dans les systèmes embarqués.

- **Bibliothèques complémentaires** : Pour faciliter la programmation et garantir la compatibilité avec les différents modules électroniques, plusieurs bibliothèques ont été intégrées [18] :
 - Wire.h : pour la communication I2C avec le module DS3231 (RTC).
 - RTCLib.h : pour simplifier la gestion de l'heure réelle avec le RTC.
 - LedControl.h ou MD_MAX72XX.h : pour l'affichage sur des matrices LED pilotées par le MAX7219.
 - Adafruit_GFX.h (si utilisation de TFT/LCD) : pour les fonctions de dessin et de texte.
 - EEPROM.h : pour stocker de manière persistante des paramètres comme la localisation ou les préférences de méthode de calcul.

Ces bibliothèques facilitent la diminution du temps de développement, l'amélioration de la fiabilité du code et la focalisation sur la logique métier, comme le calcul des horaires de prière, la gestion des affichages et l'administration du temps.

4.3. Lecture de l'heure réelle via RTC

Le module DS3231, un circuit RTC extrêmement précis, a été employé pour obtenir l'heure exacte en temps réel. Ce dispositif présente une dérive très faible (± 2 ppm) et maintient l'heure même en cas de coupure de courant, grâce à l'intégration d'une pile bouton [20].

Le microcontrôleur communique avec le DS3231 par l'interface I2C, un protocole série standardisé pour la communication entre circuits intégrés [21].

Voici les étapes principales de lecture de l'heure :

- Initialisation de la communication I2C entre le microcontrôleur et le RTC.
- Lecture des registres de temps (heures, minutes, secondes) du module DS3231.
- Conversion des données BCD (Binary Coded Decimal) en format décimal [22].
- Stockage des valeurs dans des variables pour traitement ultérieur.

4.4. Intégration de l'algorithme de calcul des horaires

Le calcul des horaires de prière est basé sur des données astronomiques comme la position du soleil [23]. L'algorithme en question se base habituellement sur des paramètres tels que :

- La latitude et la longitude du lieu.
- La date et le fuseau horaire.
- Les méthodes de calcul reconnues (UOIF, MWL, ISNA, Umm al-Qura, etc.)

Dans ce projet, un algorithme simplifié (ou basé sur une bibliothèque comme PrayTimes adaptée à l'Arduino) a été intégré. Il délivre les horaires approximatifs pour les cinq prières obligatoires : Fajr, Dhuhr, Asr, Maghrib et Isha.

Les horaires calculés sont comparés à l'heure actuelle pour déclencher des événements (comme l'affichage, l'appel à la prière, etc.).

4.5. Multiplexage de l'affichage

L'affichage des horaires de prière dans le système développé repose sur l'utilisation de modules LED à 7 segments ou de matrices LED (par exemple 8x8), pilotés par un microcontrôleur. En raison du nombre limité de broches disponibles sur les microcontrôleurs comme l'ATmega328P (Arduino Uno) ou le PIC16F877A, il est indispensable de recourir à une technique permettant de gérer plusieurs afficheurs tout en minimisant les ressources matérielles. Cette technique est le multiplexage.

4.5.1. Principe du multiplexage :

Le multiplexage d'affichage consiste à n'allumer qu'un seul chiffre ou segment à la fois, pendant une fraction de seconde, puis à passer rapidement au chiffre suivant, et ainsi de suite en boucle. Ce cycle est répété à une vitesse suffisamment élevée (généralement >60 Hz) pour que l'œil humain perçoive une image continue. Ce phénomène est connu sous le nom de persistance rétinienne [24].

Exemple :

- Pour afficher l'heure "16:30", les quatre chiffres sont affichés l'un après l'autre, chaque chiffre restant actif pendant quelques millisecondes.

- Le microcontrôleur répète ce cycle en boucle, créant l'illusion que tous les chiffres sont allumés en même temps.

Cette méthode diminue de façon significative le nombre de broches requises. Par exemple, au lieu d'avoir $7 \text{ segments} \times 4 \text{ chiffres} = 28$ connexions, on utilise seulement $7 + 4 = 11$ broches avec du multiplexage dynamique.

4.5.2. Mise en œuvre avec le circuit MAX7219

Afin de faciliter l'implémentation du multiplexage et de soulager le microcontrôleur, le projet utilise le circuit intégré MAX7219, conçu pour piloter jusqu'à 8 chiffres 7 segments ou une matrice LED 8x8. Ce circuit :

- Reçoit les données via une communication SPI (3 fils : DIN, CLK, CS).
- Gère automatiquement le rafraîchissement et le multiplexage des afficheurs.
- Permet le chaînage de plusieurs modules pour afficher plus de caractères [25].

L'architecture avec le MAX7219 présente plusieurs avantages :

- Diminution du code nécessaire dans le microcontrôleur.
- Amélioration de la fluidité de l'affichage.
- Moins de charge sur le processeur principal, libérant ainsi des ressources pour la computation des horaires et la lecture de l'heure RTC.

Le composant est contrôlé à l'aide de bibliothèques telles que LedControl.h ou MD_MAX72XX.h dans l'environnement Arduino.

4.5.3. Logique de multiplexage logiciel (sans MAX7219)

Dans les cas où le MAX7219 n'est pas utilisé, le multiplexage logiciel est mis en œuvre manuellement dans le code du microcontrôleur. Cette méthode consiste à :

- Activer une ligne d'affichage (digit) à la fois via un transistor NPN ou un ULN2003.
- Afficher les segments correspondants via des sorties numériques.
- Utiliser une temporisation (généralement via delay() ou millis()) pour passer rapidement d'un chiffre au suivant.

Le cycle d'affichage est généralement géré dans une boucle continue :

```
for (int i = 0; i < 4; i++) {  
    afficherChiffre(i, valeur[i]);  
    delay(5); // temps d'affichage de chaque chiffre
```

Cette méthode demande un calibrage précis des temporisations pour éviter le scintillement, tout en assurant une réactivité suffisante du système.

4.5.4. Intégration au système global :

Le multiplexage est intégré de manière transparente dans le programme principal, qui :

- Lit en temps réel l'heure actuelle via le RTC.
- Compare l'heure actuelle aux horaires de prière calculés.
- Affiche en continu l'heure ou l'horaire de la prochaine prière selon le mode actif.

Le système est également capable de présenter d'autres données telles que la date, le nom de la prière actuelle ou un décompte avant l'adhan.

4.6. Test et validation du système

Après la mise en œuvre de tous les modules, le système a été soumis à une série de tests pour en vérifier la fiabilité et la précision. Les tests ont inclus :

- Vérification de la précision de l'heure RTC sur plusieurs jours.
- Validation des horaires de prière en les comparant avec des sources officielles (applications mobiles ou sites spécialisés) [26].
- L'analyse de la stabilité horaire du module DS3231 pendant plusieurs jours, vérifiée avec une horloge de référence [20].
- Simulation de changement de date ou d'heure pour s'assurer de la mise à jour correcte des horaires.

Les résultats obtenus montrent que le système est fonctionnel, stable, et conforme aux attentes initiales. Les quelques écarts observés étaient marginaux et associés à la précision de l'algorithme de calcul.

Chapitre 5 :

Résultats expérimentaux et perspective

Chapitre 5 : Résultats expérimentaux et perspectives

Ce dernier chapitre présente les résultats obtenus, les points forts et les limites du système réalisé, ainsi que des pistes d'amélioration futures comme le GPS, le WiFi ou une application mobile.

5.1. Présentation des résultats

5.1.1. L'horloges temps réel (RTC) et mémoire EEPROM

Le module RTC est utilisé pour obtenir l'heure et la date avec précision et est synchronisé avec l'horloge du système. Il est connecté à la mémoire EEPROM parce que le protocole I2C permet de connecter plusieurs composants en série. La mémoire est utilisée parce que le microcontrôleur a une mémoire limitée et volatile. Nous avons donc utilisé une mémoire flash pour conserver les données et stocker les données importantes même en cas de coupure de courant.

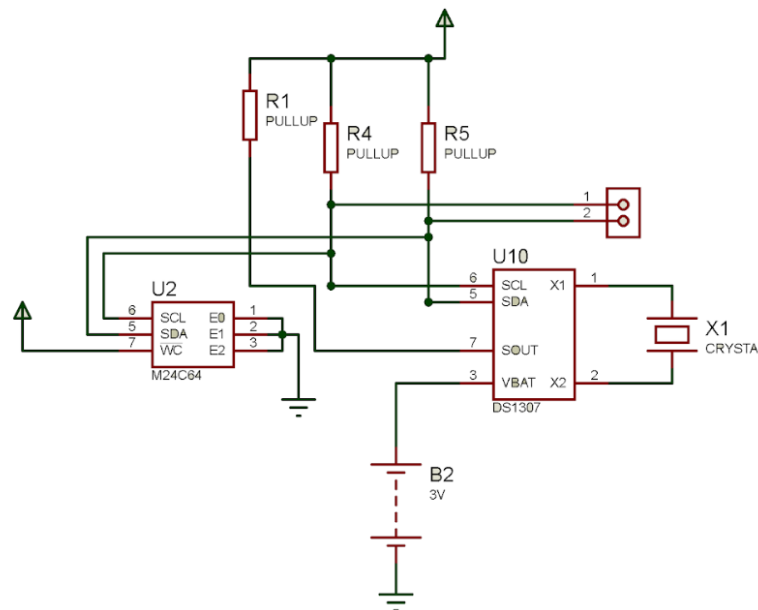


Figure 11 : Sous-circuit RTC + EEPROM

5.1.2. Les boutons de configuration

Le microcontrôleur possède des entrées et des sorties nous avons utilisé trois broches d'entrée pour connecter trois boutons dédiés à la configuration de l'heure, minutes, secondes, jour, mois et l'année en cas de problème extrême avec le système de récupération automatique de l'heure et de la date afin que l'utilisateur puisse les régler manuellement.

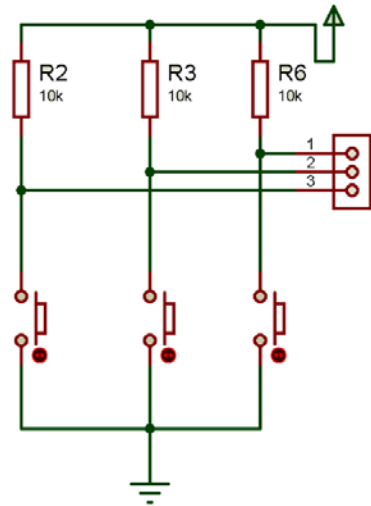


Figure 12 : Sous-circuit boutons-poussoirs

5.1.3. La date et ses composantes connectées

Le microcontrôleur lit les données du circuit RTC/Storage, les traite en données de date et d'heure qui sont transmises aux registres à décalage et en utilisant des modèles d'horloge spécifiques.

Pendant la boucle de traitement, le microcontrôleur vérifie l'entrée des boutons poussoirs et change de mode en fonction des données d'entrée.

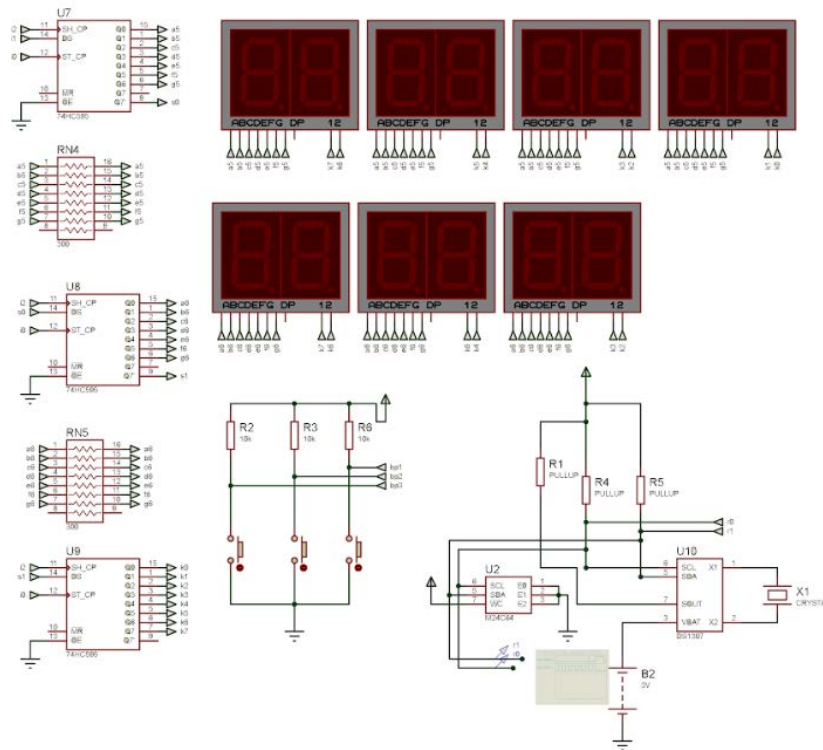


Figure 13 : Sous-circuit date et heure

5.1.4. Les temps de prière

Ce circuit est utilisé principalement pour afficher les heures de prière, il contient un module GPS et des afficheurs avec leurs registres à décalage. Le microcontrôleur lit les données de coordination du module GPS, les utilise pour calculer les heures de prière et pousse les données finales dans le registre à décalage et ensuite dans les écrans.

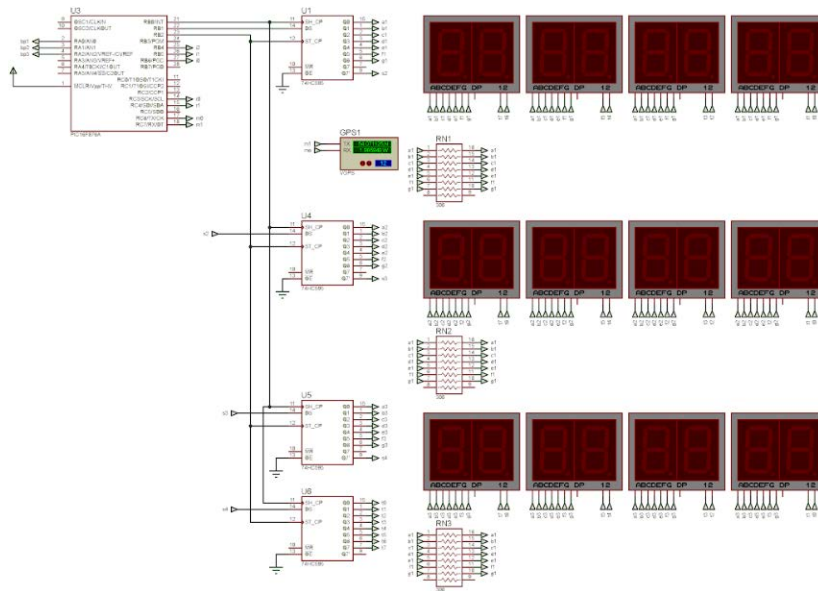


Figure 14 : Sous-circuit horaires de prière + module GPS

5.2. Limites du système

Rien ne peut être parfait et il en va de même pour notre système. Nous avons essayé d'optimiser et de résoudre tous les problèmes qui pouvaient l'être, mais certaines limites sont liées au matériel lui-même et ne peuvent donc pas être corrigées. Nous avons trouvé les problèmes suivants :

- Capacité d'affichage limitée : Les afficheurs 7 segments ne permettent que la représentation de chiffres et quelques lettres. Pour des applications nécessitant des affichages graphiques avancés, un écran LCD ou OLED serait plus adapté.
- Sensibilité aux perturbations électromagnétiques : Les systèmes embarqués peuvent être sensibles aux interférences électromagnétiques, ce qui peut perturber les signaux de transmission.
- Contraintes de stockage et de traitement des données : En raison de la taille et des spécifications du microcontrôleur, le système ne peut pas contenir plus d'un certain nombre

de coordinations si l'on utilise des numéros préconfigurés et il ne peut pas effectuer de traitement massif des données, de sorte que vous pouvez constater des ralentissements de temps à autre.

- Adaptabilité du système à des évolutions : La mise à jour de nouvelles fonctionnalités ou méthodes de calcul nécessite une refonte complète du système.

5.3. Améliorations possibles

5.3.1. GPS

L'intégration d'un module GPS permet au système d'être portable et de s'installer facilement dans n'importe quel endroit grâce à la récupération en temps réel de la géolocalisation qui donne plus de précision dans les calculs de prières.

5.3.2. WiFi

L'intégration d'un module WiFi permet au système d'obtenir des données de localisation approximatives qui peuvent être utilisées pour calculer les temps de prière, synchroniser l'horloge du système avec le temps universel coordonné (UTC), les mises à jour du système, les temps de prière pré-calculés et les informations météorologiques.

5.3.3. Affichage TFT

Un écran TFT améliore la lisibilité du système et permet d'afficher des graphiques avancés et des informations plus dynamiques, avec la possibilité d'animer l'interface utilisateur, ce qui donne au système une impression de modernité.

5.3.4. Application Mobile

Une application pour smartphone peut être créée pour ce système afin de le configurer et de le contrôler à distance ou, par exemple, de recevoir des alertes ou de visualiser les données enregistrées.

Conclusion Générale

Conclusion générale

Conclusion générale

Ce projet décrit le fonctionnement d'un système qui affiche les données de l'horloge en temps réel ainsi que les heures de prière. Il utilise un microcontrôleur PIC 16F876A, des registres à décalage, des écrans à 7 segments et d'autres modules tels que le GPS.

Il lit l'horloge en temps réel du module RTC, les données de géolocalisation du module GPS et les traite à l'aide du microcontrôleur qui transmet ensuite les données aux registres à décalage en utilisant leur modèle d'horloge spécifique, qui est ensuite transmis aux écrans, tout en vérifiant l'entrée des trois boutons de configuration : sélection, haut et bas.

Ce projet peut être amélioré grâce à des technologies de pointe qui lui donneront un aspect plus moderne.

Références Biographiques

- [1] site internet https://fr.wikipedia.org/wiki/Syst%C3%A8me_temps_r%C3%A9el
- [2] site internet <https://www.techno-science.net/definition/7739.html>
- [3] site internet <http://www.ordinateur.cc/Mat%C3%A9riel/Durs-et-stockage/48566.html>
- [4] site internet <https://www.orientdisplay.com/fr/knowledge-base/lcd-basics/what-is-lcd-liquid-crystal-display/>
- [5] site internet <https://www.tme.eu/fr/news/library-articles/page/63586/lcd-tft-led-oled-et-autres-a-propos-des-affichages-electroniques/>
- [6] site internet <https://www.electricity-magnetism.org/fr/affichages-a-sept-segments/>
- [7] site internet <https://cictro.com/tutoriels/blog/7-segment-display-tutorial-html/>
- [8] site internet <https://cosmo-games.com/technologie-oled/>
- [9] site internet <https://umvie.com/decouvrez-le-fonctionnement-de-lecran-oled-explications-claires-et-concises/>
- [10] site internet <https://www.guideavis.com/quest-ce-que-la-technologie-oled-et-comment-fonctionne-t-elle/>
- [11] site internet <https://fr.hwlibre.com/Guide-complet-du-module-RTC-DS1307/>
- [12] site internet https://fr.wikipedia.org/wiki/Horloge_temps_r%C3%A9el
- [13] site internet <https://www.analog.com/media/en/technical-documentation/data-sheets/DS1307.pdf>
- [14] site internet <https://www.electricity-magnetism.org/fr/registres-a-decalage/>
- [15] site internet <https://www.abcelectronique.com/annuaire/cours/cache/1155/le-pic-16f84.pdf>
- [16] Arduino Documentation. <https://www.arduino.cc/en/Guide/Environment>
- [17] MikroElektronika. MikroC PRO for PIC - User Manual. <https://www.mikroe.com/mikroc/pic>
- [18] Adafruit Libraries for Displays. <https://learn.adafruit.com>

- [19] Maxim Integrated, DS3231 Datasheet www.datasheets.maximintegrated.com/en/ds/DS3231.pdf
- [20] NXP Semiconductors, UM10204 - I2C-bus specification and user manual.
- [21] Embedded Systems Programming in C and Assembly, Adam Taylor.
- [22] Jean Meeus, Astronomical Algorithms, 2nd Edition, Willmann-Bell, Inc., 1998.
- [23] Donald Comer, *Digital Design with CPLD Applications and VHDL*, Thomson, 2003.
- [24] MAX7219 Datasheet, Maxim Integrated. <https://www.analog.com/media/en/technical-documentation/data-sheets/MAX7219-MAX7221.pdf>
- [25] IslamicFinder.org – Accurate Prayer Times & Qibla Direction. <https://www.islamicfinder.org>
- [26] Figure Configuration D'afficheur 7 Segment <https://labs.dese.iisc.ac.in/embeddedlab/tm4c123g-launchpad-7-segment-led-interface/>
- [27] Figure Brochage D'un DS1307 <https://www.watelectronics.com/ds1307-rtc-module/>
- [28] Figure Brochage D'un 74HC595 <https://learn.adafruit.com/74hc595/pinouts>
- [29] Figure Brochage D'un PIC16F876 <https://commons.wikimedia.org/wiki/File:Pic16F876.png>
- [30] Figure Dimension d'un double afficheur 7 segment <https://docs.broadcom.com/doc/AV02-3586EN>
- [31] Prayer Times Calculation <https://praytimes.org/calculation>