B.1.1) RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

B.1.2) UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE

Mémoire

En vue de l'obtention du diplôme de

Master en Informatique

Option : Réseaux, Mobilité et Systèmes Embarqués

Thème

Système de détection d'intrusion basé sur un processeur ARM

Réalisé par :
Mr. HAMMICHE Abdelmalek
Mr. HADDAD Nacer

Promoteur :
Mr DAOUI Mehammed

Co-promoteur :
Mr DJIOUA Smail

Promotion : 2013-2014

Sommaire

Introduction général

Introduction

La sécurité revêt une importance primordiale pour tout être humain. Aujourd'hui les nouveaux obstacles auxquels fait face le sujet, ont fait naitre de nouveaux dispositifs sécuritaires, en vue de parer à toute menace venant de ces contraintes. A partir de là, l'idée de la surveillance à distance a été un concept plus qu'intéressant pour s'armer contre toute nuisance menaçant ses biens.

De nombreux systèmes de surveillance ont vu jours. De simples systèmes mécaniques (cloche derrière la porte) à des systèmes électriques (ex : relais électrique) jusqu'à des systèmes sophistiqués. Ce dernier point fait référence à l'émergence des systèmes embarqués dans le domaine sécuritaire. Rendant possible cette surveillance à distance, les systèmes embarqués font place à un large espace de fonctionnalités éventuelles.

Notre objectif dans ce projet se résume par une conception d'un système embarqué pour la surveillance de biens privé.

Ce système de surveillance, conçu à base d'un microcontrôleur (pour rependre aux aspects d'intelligence), entouré par des modules périphériques tels que, des capteurs, des ILS et un clavier (pour la collection de données) d'une parte, une sirène et un module réseau (pour la signalisation) d'autre part, et enfin, une interface utilisateur à travers un afficheur LCD (information sur état).

Ce mémoire est formé de quatre chapitres, à travers lesquels, nous décrivons le travail effectué pour la conception et la réalisation de notre système:

Dans le premier chapitre, nous présentons les généralités sur les systèmes embarqués, dans le second chapitre nous décrivons les généralités sur les systèmes d'alarmes, dans le chapitre 3 nous nous étayerons sur la phase de conception de notre système de surveillance. On y décrit les principaux composants constituants ce système, tout en justifions leurs choix, dans le chapitre 4 nous parlerons de la mise en pratique de notre conception.

Chapitre 1

Généralités sur les systèmes embarqués

I. 1) Introduction

L'essor exponentiel du monde de l'électronique durant les dernières décades, n'a pas été vain pour l'entourage de l'être humain. Exploité en masse par et pour ce dernier, et rendant omniprésents des composant électroniques dans sa vie, notamment grâce à une grande miniaturisation du système, rares sont les domaines qui n'ont pas été affectés. Appelés fréquemment systèmes embarqués ou systèmes enfuis, ces composants électroniques sont des ordinateurs basés systèmes, avec une vision différente des ordinateurs standards

I. 2) Définitions des systèmes embarqués

- **Système embarqué**

Dans le domaine informatique, un système embarqué est un système comportant une partie matérielle (hardware) et une autre partie logicielle (software). Ces deux parties interagissent entre elles pour assurer une ou plusieurs fonctions définies (comme illustré dans la figure 1), ayant comme but l'amélioration de l'environnement technique. Cela peut se traduire par la fourniture d'un système de contrôle sophistiqué, qui lui ajoute des fonctionnalités supplémentaires, et/ou facilite le fonctionnement du système [1].

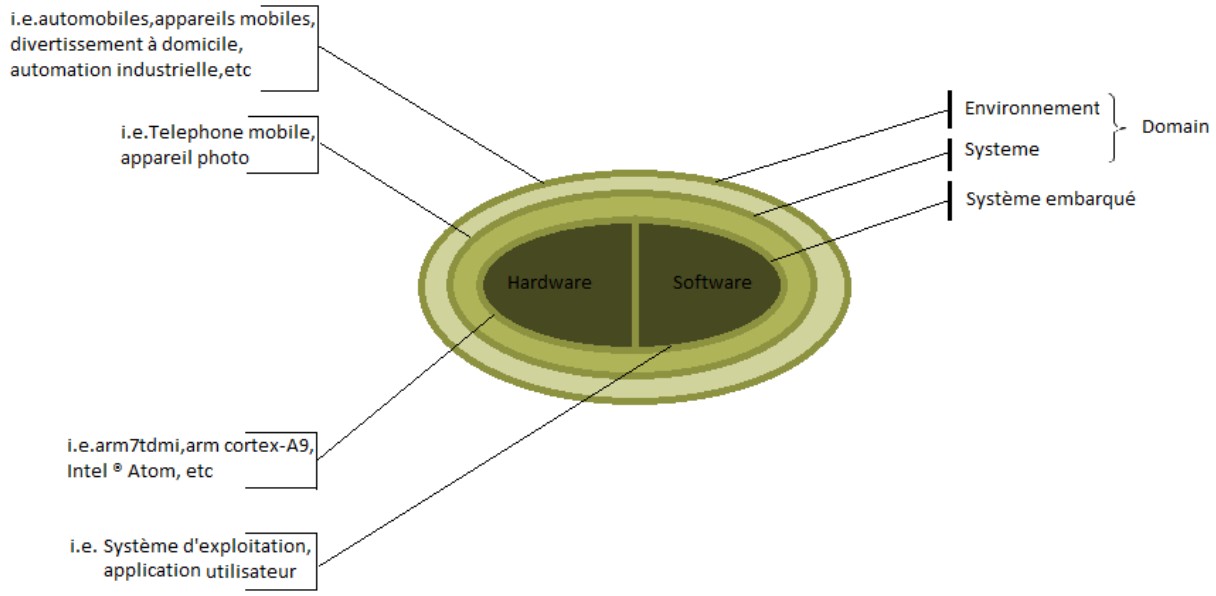


Figure 1 synoptique d'un système embarqué

Un système embarqué fonctionne généralement en temps réel, mais une faible barrière existe entre systèmes embarqués et systèmes temps réel, du fait que le logiciel embarqué peut ne pas présenter des contraintes temps réels.

L'enfouissement de la partie logiciel rend ce dernier sûr, ce qui implique une très grande fiabilité du système.

I. 3) Historique

La recherche systématique de la facilité, de l'assurance, et de la rapidité a toujours été au centre des préoccupations de nos prédécesseurs. Ces derniers ont toujours aspiré à la simplification du mode de vie en inventant des systèmes autonomes, quoique mécaniques (l'horloge mécanique : gnomon, machine arithmétique de blaise pascal en 1643). Ces systèmes mécaniques ont toujours eu comme inconvénients la taille et le temps d'exécution.

L'évolution électronique qu'a connue le monde a eu un impact significatif dans l'amélioration des contraintes citées. L'invention du tube à vide au début du vingtième siècle, a joué un rôle majeur dans la pensée pré-systèmes embarqués. Cette invention a donné naissance aux premiers postes radios et à l'apparition des postes de télévision. À leurs apogées, les tubes à vide ont contribué à l'invention des premiers ordinateurs, d'abord électromécanique pour citer le z1etz3 en 1938[2], Harvard marque en 1944[3], pour arriver à une conception purement électronique (ENIAC en 1946 avec 17468 tubes à vide et un poids de 30 tonnes [4]). L'avancée significative par l'apparition du transistor qui a aussitôt balayé les tubes à vide dans l'électronique a permit un gain significatif en performance et en taille. Les circuits intégrés ont quant à eux donné naissance au premier système embarqué proprement dit, et qui n'est autre que celui présent dans le fameux projet Apollo, pesant 70 kg et qui amena N. Armstrong à la lune. C'est à partir de là que commença l'idée des systèmes embarqués. La première production en série d'un système embarqué est vraisemblablement le D17 Autonetics qui servait de système de contrôle aux missiles nucléaires américains [5]. Une multitude de systèmes embarqués a vu le jour depuis, nous citerons quelques-uns ci-après:

- 1972 Premier microprocesseur 8 bits (48 instructions, 800kHz) intitulé Intel 8008.
- 1974 Premier microprocesseur largement diffusé 8 bits, (64KB d'espace adressable, 2MHz - 3MHz) intitulé 8080 d'Intel
- 1978 Création du Z80, processeur 8 bits.
- 1979 : création du MC68000, processeur 16/32 bits

Ces systèmes embarqués ont pris place graduellement dans notre vie. Ils sont devenus omniprésent et indétrônables à partir des années 1980. Le seul moyen de les différencier est

une classification pertinente qui est directement conséquente des facteurs de conception considérés.

I. 4) Classification des systèmes embarqués

I.4.1. Système Transformationnel

Outre son activité de calcul, il lit ses données et ses entrées lors de son démarrage puis fournit ses sorties, et finalement meurt.

I.4.2. Système Interactif

Système en interaction quasi permanente avec son environnement, y compris après l'initialisation du système. La réaction du système est déterminée par les événements reçus et par l'état courant (en fonction des événements et des réactions passés). Le rythme de l'interaction est déterminé par le système et non par l'environnement.

I.4.3. Système Réactif ou Temps Réel

Système en interaction permanente avec son environnement, y compris après l'initialisation du système. La réaction du système est déterminée par les événements reçus et par l'état courant (en fonction des événements et des réactions passés); mais le rythme de l'interaction est déterminé par l'environnement et non par le système.

I. 5) Facteurs de conception des systèmes embarqués

Le succès qu'a connu l'utilisation des systèmes embarqués, a fait que ces derniers se sont implantés dans divers domaines. Ce qui engendre une variation dans les facteurs de conception à prendre en considération d'un système à concevoir a un autre. Leurs utilisateurs, les ressources disponibles, la durés de vie, tous lié à l'environnement sensé abriter (accueillir) le système embarqué, etc...[6]

1.5.1. Environnement

L'environnement est sans aucun doute le facteur le plus essentiel parmi les facteurs de conception existants, principalement à cause de son hostilité du fait des différentes contraintes qu'il nous impose.

a. Convivialité

L'utilisateur ne veut généralement pas se former ou se contraindre à consulter un manuel pour utiliser le système, bien que cela s'applique à d'autres systèmes informatiques. La restriction avec les systèmes embarqués est que le fonctionnement du système lui-même n'est très probablement pas la principale tâche des utilisateurs, comme par exemple le traitement d'un système multimédia dans une automobile, bien que l'accent mis par l'utilisateur est défini sur le trafic routier. Les interfaces graphiques ou interface utilisateur permettent d'améliorer ces systèmes, idéalement couplées avec diverses possibilités pour les entrées des utilisateurs (par exemple, reconnaissance de la parole, écrans tactiles, boutons et de cadrans) et le système de sortie (par exemple text-to-speech « lecture auditive d'un texte», Head-Up-display «viseur tête haute »).

b. Climat

En raison du fait que les systèmes embarqués sont exploités dans des environnements très divers, les exigences en matière de conditions climatiques prises en charge varient encore plus. Ceci inclut par exemple la température, l'humidité, la pression atmosphérique et la pression de l'eau. Les implications associées concernent essentiellement la partie matérielle de ces systèmes.

c. Durée de vie

La durée de vie est liée étroitement à la partie hardware du système. Pendant la durée de vie d'un système embarqué l'accès pour la maintenance du système est très probablement limité, voire impossible (La sonde spatiale Pioneer 10 lancée en 1972 par la Nasa avec un dispositif hardware rependant aux normes des systèmes embarqué, équipée du processeur Intel 4004 ne peut atteindre son objectif que d'ici deux million d'années). Ajouté à cela les coûts de rappel pour un produit défaillant qui, combiné avec l'aspiration au moindre cout possible (facteur économie ci-après), donne une concurrence déloyale. Cela implique que le système déployé doit être très bien testé, et doit être pleinement opérationnel tel que requis par le client du système à travers son cahier de charge.

d. Mobilité

Souvent les systèmes embarqués sont intégrés dans les systèmes qui ne sont pas liés à des lieux géographiquement statiques. Des petits lecteurs MP3 jusqu'aux vitres électriques des automobiles ou à l'unité de moteur d'un navire porte-conteneurs de contrôle. Le système peut être appelé à prendre en compte des exigences implicites telles que le support de coups et de chocs pendant le fonctionnement.

e. Économie

Les systèmes embarqués se trouvent dans un marché très concurrentiel, et cette donne fait que le sujet est soumis à des mesures d'économie plus que dans d'autres systèmes informatiques. Pour l'emporter, un producteur doit non seulement accélérer le temps de mise sur le marché « time-to-market» mais aussi maintenir le prix par unité au minimum.

Pour pouvoir modéliser un système embarqué, aucun aspect ne doit être considéré comme distinct des autres. Une bonne connaissance du domaine ainsi qu'une bonne maitrise du système à concevoir sont nécessaires afin de réaliser un système utilisable projeté au déploiement. Par conséquent la connaissance suffisante du domaine est nécessaire ainsi qu'une excellente connaissance du système. Des normes ont été élaborées pour rendre la capacité de gérer l'environnement et ses nuances comparable. Le code Indice de protection (IP), tel que défini dans la norme internationale IEC60529, classe la protection prévue contre la pénétration de corps solides, poussières, contact accidentel, ainsi que l'eau dans les armoires électriques.

1. 5. 2 Matériel

Pour tenir compte des incidences de l'environnement tel que présenté ci-dessus, le matériel doit être choisi avec soin lors de la conception d'un système embarqué. Cela doit aussi inclure la prévoyance pour fournir une puissance de calcul suffisante pour le problème à résoudre en utilisant le logiciel, qui sera discuté dans la section suivante.

En général, un système informatique est constitué de la saisie de données, des composants de traitement de données, et enfin de la sortie des données traitées. Ce paradigme ne change pas pour les systèmes informatiques embarqués. Le système des entrées/ sorties est réalisé par divers capteurs et/ou actionneurs placés dans un système technique entourant alors les unités d'entrées/sorties d'un système informatique (ex : un microcontrôleur) qui se charge du traitement des données, quant aux interconnexions à incorporer au système informatique, elles sont réalisées par tout dispositif communicatif (ex : module wifi supportant la communication par onde électromagnétiques). La communication filaire est supportée par l'ensemble des systèmes informatiques à cet égard (ex : USART).

On peut comprendre alors que la tâche principale d'un système embarqué est réalisée par un système informatique et qui est vraisemblablement un processeur. Les autres périphériques

notamment les capteurs, n'ont de rôles que comme collecteurs de données. Bien que cet objectif n'est pas un facteur négligeable, et on pourrait dire au passage que tous ces périphériques doivent être fiables. Nous nous intéresserons qu'aux systèmes informatiques et aux processeurs proprement dit.

Lors de l'évaluation des processeurs, il est important de comprendre la différence entre les architectures de jeux d'instructions (ISA) et une réalisation de processeur ou une puce. Cette dernière est un morceau de silicium à semi-conducteur. La première est une définition des instructions que le processeur peut exécuter et certaines contraintes structurelles (comme la taille de mot) que les réalisations doivent partager ; Comme par exemple l'architecture RISC ou CISC. Il y a beaucoup de réalisations. Une ISA est une abstraction partagée par de nombreuses réalisations. Une seul ISA peut apparaître dans de nombreuses puces différentes, souvent faites par différents fabricants, et ayant souvent des profils de performances très variables [7].

En usage informatique général, la variété des architectures du jeu d'instructions(ISA) est aujourd'hui limitée, l'architecture Intel x86 domine massivement. Cela dit la notion de dominance dans l'informatique embarquée n'a pas de place. Au contraire, la variété des architectures peut être intimidante pour un concepteur de Système embarqué.

L'avantage de partager un ISA dans une famille de processeurs est que les outils logiciels, qui sont coûteux à développer, peuvent être partagés, et (parfois) les mêmes programmes peuvent fonctionner correctement sur plusieurs réalisations. Cette dernière propriété, cependant, est assez perfide, car un ISA ne comprend pas normalement les contraintes du timing. Par conséquent, même si un programme peut s'exécuter logiquement de la même façon sur des puces multiples, le comportement du système peut être radicalement différent lorsque le processeur est incorporé dans un système de cyber-physique.

Lorsqu'ils sont déployés dans un produit, les processeurs intégrés, assurent généralement une fonction dédiée ; ils contrôlent un moteur d'automobile ou ils mesurent l'épaisseur de la glace dans l'Arctique... Ils ne sont pas invités à remplir des fonctions arbitraires avec le logiciel défini par l'utilisateur. Par conséquent, les processeurs peuvent être très spécialisés. Ce qui les rend plus spécialisé peut apporter des avantages en matière de consommation d'énergie, et par conséquent être utilisables avec de petites batteries pour de longues périodes

de temps. Ils peuvent aussi inclure du matériel spécialisé pour effectuer des opérations qui seraient coûteuses à effectuer sur le matériel à des fins générales, telle que l'analyse de l'image.

a. Types de processeurs

Un processeur ou CPU est le composant qui exécute les instructions machine des programmes informatiques.
En conséquence de la grande variété d'applications embarquées, il y a une grande variété de processeurs qui sont utilisés. Elles vont de la très petite, lentes, peu coûteuses, appareils de faible puissance, à haute performance à des dispositifs spéciaux. Cette section donne un aperçu de quelques-uns des types de processeurs disponibles.

- **Microprocesseur :**

Un microprocesseur est un processeur contenu dans un seul circuit intégré. Composé de trois unités majeures qui sont, l'unité arithmétique et logique (UAL), l'unité de commande et des registres.

L'unité de commande permet de "séquencer" le déroulement des instructions. Elle effectue la recherche en mémoire de l'instruction, le décodage, l'exécution et la préparation de l'instruction suivante.

L'unité UAL est associée à deux registres, l'accumulateur et le registre d'état (flag ou drapeau). Elle effectue les opérations arithmétiques (addition, multiplication,…) et logique (AND, OR, XOR,…) en se servant des accumulateurs. Le registre d'état contient une indication sur l'opération effectué (retenu, signe +/-, débordement,..).

Les registres se composent en deux. Les registres généraux qui sont des mémoires rapides, à l'intérieur du microprocesseur, qui permettent à l'UAL de manipuler des données à vitesse élevée. Ils sont connectés au bus de données interne au microprocesseur. Les registres d'adresse sont connectés au bus d'adresse comme le compteur ordinal (PC), le pointeur de pile (SP).

Exemple des microprocesseurs :

x86 (Intel, AMD), PowerPC (IBM, Motorola), SPARC, MIPS ARM (systèmes embarqués)

- **Microcontrôleurs**

Un **microcontrôleur** (en notation abrégée **µc**, ou **uc** ou encore **MCU** en anglais) est un circuit intégré qui rassemble les éléments essentiels d'un ordinateur : processeur, mémoires (mémoire morte pour le programme, mémoire vive pour les données), unités périphériques et interfaces d'entrées-sorties[8].

A l'origine, les microcontrôleurs avaient des mots de 8 bits pour proposer des avantages tels que la faible consommation d'énergie, le bas cout et une dimension réduite pour bien s'adapter à des applications qui nécessitent de petites quantités de mémoire et des fonctions logiques simples. En passant par des processeurs 16 bits qui, cependant se sont relativement peut rependus. La tendance actuellement est de s'éloigner de plus en plus des microcontrôleurs 8 bits pour faire place aux UC 32 bits et ce essentiellement à cause des prix plus qu'intéressants des processeurs 32 bits (le processeur 32 bits le moins chère peut ne couter qu'un seul euro) ; Proposant des avantages tels que puissance et vitesse de calcul supérieure avec une taille de la mémoire relativement grande, pour un espace d'adressage plus grand, capacité d'héberger un système d'exploitation (grâce la performance CPU et la taille mémoire). Généralement la taille requise pour les UC dans les SE s'étend de quelques kilos octets a des centaines de kilos octets pour avoisiner quelques Méga octets pour ceux sensés abriter un système d'exploitation. Toute en proposant des fonctions arithmétiques exigeantes en performances, ces processeurs sont conçus pour utiliser relativement peu d'énergie sans sacrifier les performances par rapport aux processeurs utilisés dans les ordinateurs de haut de gamme, comprennent souvent un mode en veille qui permet de réduire la consommation d'énergie à quelques nW(nano watts) seulement. Des composants embarqués tels que les nœuds de réseaux de capteurs et des dispositifs de surveillance ont été mis en évidence pour fonctionner avec de petites batteries pour plusieurs années.

- **DSP Processeurs**

Les DSPs ou les processeurs de traitement de signal sont renommés pour être utilisés dans le traitement de signal. Initialement ils sont utilisés pour gérer la Carte son d'un ordinateur. Les DSPs ont vu leurs utilisations s'accroître considérablement depuis 1985, tout d'abord grâce au développement des télécommunications (téléphonie numérique, puis téléphonie sans fil GSM ...), puis grâce à leurs possibilités de traitement rapide de certaines commandes numériques faisant appel à des algorithmes complexes permettant ainsi le travail en « temps réel ».

Grâce à leur architecture qui est optimisée pour traiter une grande quantité de données en parallèle à chaque cycle d'horloge, les DSPs sont célèbres par leur calcul numérique intensif, leurs précisions de calcule numérique, le débit mémoire important, et leurs capacités de fonctionnement en temps réel. Leurs possibilités de traitements de signal sont réalisées de manière qu'il sera impossible à le faire avec les processeurs classiques [9]. Par exemple si nous prenons l'équation suivante : $A \leftarrow A + (X.Y)$, la différence de traitement est illustrée ci bas :

Processeur classique :

Accès mémoire X puis Y a 2 cycles
Addition a 1 cycle
Multiplication a 1 cycle
} 4 cycle

Processeur DSP :

Accès mémoires + MAC à 1 cycle } 1 cycle

Toutes les applications de traitement du signal partagent certaines caractéristiques. Tout d'abord, elles traitent de grandes quantités de données. Les données peuvent représenter des échantillons dans le temps d'un processeur (tels que des échantillons d'un signal radio sans fil), des échantillons dans l'espace (comme les images), ou les deux (comme la vidéo et radar). Deuxièmement, elles effectuent généralement des opérations mathématiques complexes sur les données, y compris le filtrage, l'identification du système, l'analyse de fréquence, l'apprentissage machine. Ces opérations sont mathématiquement intensives.

Ces processeurs étant conçus spécifiquement pour soutenir le calcul numérique intensif des applications de traitement du signal, leur tendance est de traiter un programme spécifique, et non la totalité des programmes qui sont appelé à être traités ; de ce fait dans plusieurs cas les DSPs sont ajoutés en tant que modules, pour traiter un programme tel que l'algorithme de filtrages, transformation temps en fréquence, corrélation….

Aujourd'hui parmi les domaines ou les DSPs sont les plus utilisés, nous citons la télécommunication comme dans les Modem, ou les multiplexeurs, les interfaces vocaux avec la reconnaissance automatique de la parole et la synthèse vocale, le domaine militaire incluant le guidage de missiles, le MultiMedia compression des signaux audio, carte multi Media pour

ordinateur, le domaine médical comprenant la compression d'image médicale (IRM échographie),…

- **Processeur graphique GPU**

L'unité de traitement graphique (GPU) est un processeur dédié au traitement des données graphiques. Ces processeurs remontent aux années 1970, Avec l'augmentation gigantesque de leurs performances, les GPU peuvent maintenant effectuer une pléthore de calculs parallèles. D'abord utilisé à des fins de décryptage, l'ancien compagnon du CPU entre en concurrence directe avec son comparse et s'immisce dans l'industrie des superordinateurs. Certaines applications embarquées, notamment des jeux, sont un bon conteste pour les GPU. De plus, les GPU ont évolué vers des modèles plus généra aux de programmation, et donc ont commencés à apparaître dans d'autres applications de calcul intensif, comme l'instrumentation (accélérateur de particules). Les GPUs sont généralement assez gourmands en énergie, et donc aujourd'hui ne sont pas un bon chalenge pour les applications embarquées à contraintes énergétiques. Les principaux fournisseurs de GPU d'aujourd'hui sont Intel, NVIDIA et AMD [7].

Aujourd'hui nous sommes témoins de l'apparition du haut degré d'intégration, on peut trouver des systèmes embarqués avec un système informatique central (un processeur, microcontrôleur) ; gérer d'autres systèmes informatiques dédiés, notamment des DSPs pour le traitement de signal, et/ou des GPUs pour leurs tâches spécifiques.

1. 5. 3 Software

A l'apparition des systèmes embarqués, la partie logiciel accompagnant ce dernier était en fait constituée d'un seul programme, principalement dû aux conceptions simplistes de ces systèmes embarqués. L'apparition de hauts degrés d'intégration permet d'envisager des circuits pouvant contenir plusieurs centaines de milliers de portes logiques a fait que le software a connu une ascension fulgurante jusqu'à l'apparition des OS dans les systèmes embarqués.

a. Langages de programmation dans les systèmes embarqués

Le logiciel des systèmes embarqués doit tenir compte des ressources et des capacités du matériel disponible qui est limité, et prendre en compte les circonstances de l'environnement tel que décrit dans les sections précédentes. Ceci implique une utilisation efficace de la puissance de calcul et de la mémoire. Par conséquent, une solution logicielle est mise en

œuvre en utilisant un langage de programmation, qui offre un haut degré de liberté lié à la consommation mémoire et l'utilisation efficace de la capacité de calcul pour le développeur du système ; Raison pour laquelle l'assembleur a encore le droit d'exister dans le monde de l'embarqué. Des langages comme C et C++offrent également un contrôle similaire à son développeur, mais à un niveau plus élevé d'abstraction, offrant un gain en maintenabilité et en portabilité. Le C/C++ peut fournir des capacités à intégrer des parties de code assembleur dans les fonctions, soutenir le développeur pour accéder aux ressources du système directement ou mettre en œuvre des algorithmes de temps critiques. L'avantage des langages de programmation tel que C/C++ est la réduction des coûts de développement, ce qui revient à améliorer le temps d'accès au marché.

- **Programmation orientée objet**

Apparu après la programmation assembleur et la programmation impérative, la programmation par objet constitue un pas de plus dans la capacité du langage à s'abstraire du langage machine toute en augmentant la portabilité. Pas toujours commode, l'orienté objet a beaucoup plus percé dans le marché des télécommunications mobiles ainsi que dans le système d'exploitation mobile propagé de Google pour les appareils embarqués Android. Comme langage on peut citer java, en particulier la plate-forme Micro Edition (ME). Pour pouvoir supporter l'orienté objet, le système se doit d'abriter un OS.

b. Programmation des logiciels embarqués

Les systèmes informatiques, avant l'apparition du concept de système d'exploitation, étaient gérés par des simples bibliothèques, écrites en même temps que le développement des applications et du matériel. Cependant une telle approche engendre des difficultés lors du développement de systèmes complexes car le système embarqué doit être conçu dans son ensemble, d'un seul coup. Les temps de conception sont ainsi très longs.

L'apparition de méthodes de Co-design hardware/software a grandement contribué à la facilité de développement de logiciel embarqué. Basé sur la modélisation à haut niveau de spécification, la méthode Co design fait qu'à la fois les parties logicielles et matérielles vont être synthétisées.

Une autre approche pour la partie logicielle est l'titillation d'une couche logicielle servant de système d'exploitation sur la partie matérielle. Les applications s'exécutent alors au-dessus de cette dernière, en présentant de nouveaux aspects non déjà rencontrés dans des logiciels standards.

c. Caractéristique d'un logiciel embarqué

Prendre en considération les contraintes de la partie hardware, rend très spécifique le logiciel embarqué ; le logiciel embarqué se doit :

1. d'être discret : il doit savoir se faire oublier.

2. d'être ciblé : limité aux fonctions pour lesquelles il a été créé.

3. d'être fiable et sécurisé : destiné à un fonctionnement autonome et/ou critique.

4. d'avoir une longue durée de vie : ex : dans le domaine spatial.

5. d'avoir des IHM (interactions homme machine) spécifiques ou déportées :
   - Affichage réduit ou inexistant
   - Pas forcément de clavier/souris
6. d'être optimisé: empreinte mémoire, performances
   - gagner quelques euros « € » sur le matériel peut être important
   - processeurs (encore) moins puissants que dans un ≪ PC ≫
   - temps d'attente peu acceptable pour l'utilisateur non spécialiste
7. Pas obligatoirement un OS

- Importance du couple matériel logiciel :

L'importance du couple cité est très palpable dans les systèmes n'ayant pas de MMU alors pas de protection mémoire. De plus la majorité des OS n'utilisent pas cette MMU par défaut. Le logiciel doit être fait de manière qu'il n'y est pas de place à l'erreur causant une impasse pour le système.

I. 6) Système d'exploitation dans les systèmes embarqués

Afin d'améliorer la portabilité des systèmes et faire bon usage des pilotes déjà disponibles pour les composants matériels sélectionnés, il peut être nécessaire de faire appel à un système d'exploitation(OS) de forme abstraite de la plate-forme sous-jacente. Généralement offrant des capacités en temps réel, dans cette section nous citerons quelques systèmes d'exploitation basé sur une plate-forme matérielle fournissant une unité de gestion de mémoire(MMU).

Rôle des systèmes d'exploitation dans l'embarqué

Ayant comme but d'affranchir le développeur de logiciel embarqué de la maitrise du hardware, ce qui engendre une répercussion positive sur le temps de développement et alors le time-to-market. Les systèmes d'exploitations offrent les possibilités de bénéficier des mêmes avancées technologiques que les applications classiques (TCP/IP, HTTP…) toute en augmentant la portabilité du système quoiqu'au détriment de l'empreinte mémoire.

Exemple des OS embarqués

• **VxWorks** : noyau RT le plus utilisé dans l'industrie. Supporte TCP/IP et une API socket.
• **QNX** : noyau RT de type UNIX. Intègre une interface graphique proche de XWindow (Photon). QNX peut être utilisé gratuitement pour des applications non commerciales. Très faible empreinte mémoire.
• **uC/OS** : destiné à des microcontrôleurs type Motorola 68HC11. Support de TCP/IP. Gratuit pour l'enseignement.
• **RTEMS** : Real-Time Operating System for Multiprocessor Systems. Gratuit porté par de nombreuses architectures
• **Windows CE** : Victime d'une réputation de fiabilité approximative...
• **Lynx OS** : Système RT conforme à la norme POSIX[1].
• **Nucleus** : Noyau RT avec support de TCP/IP, une interface graphique, un navigateur web et serveur HTTP. Livré avec les sources, pas de royalties à payer pour la redistribution.
• **eCos** : Noyau RT *(Embeddable Configurable OS)*, faible empreinte mémoire, basé sur Linux et la chaîne de *cross-compilation* GNU (POSIX). Support de TCP/IP. Licence proche GPL. Largement utilisé (automobile, imprimante, lecteur MP3).
Support des processeurs x86, PowerPC, SH3 Hitachi ou StrongARM.

I. 7) Caractéristiques des systèmes embarqués

Venir à bout des exigences environnementales des systèmes embarqués, fait naitre des caractéristiques propres à ces systèmes embarqués. Nous résumons ces caractéristiques en ces quelques points suivant :

- **Le non flexibilité**

Les systèmes embarqués ne sortent jamais de leur cibles et ce, principalement pour ne pas répondre a d'autre exigence qui impacteront le cout et le time-to-market (rependre au facteur économie).

- **Temps de conception**

[1] POSIX norme de standardisation des interfaces de programmation des logiciels

Les concepteurs de systèmes embarqués se doivent de concevoir leurs produis en un temps relativement cours pour répondre aux besoins du marché.

- **Critère de performance**

Différents des autres systèmes informatiques, dans les systèmes embarqués les critères coût, fiabilité et sécurité priment souvent sur la performance.

- **Interface utilisateur**

Bien qu'elle ne joue pas le rôle d'interface utilisateur standard du fait qu'elle ne représente pas un circuit d'entré (pas de permission de manipulation de la machine) (quoique), elle ne constitue que des entrées de donnée. Bon nombre d'interfaces utilisateur ont faits surface pour augmenter la Convivialité.

Quoique : si les afficheurs LCDs ou autre n'ont pas de fonction permettant de les manipuler physiquement (pour communiquer on doit nécessairement proposer un clavier ou autre circuit d'entré), on trouve des afficheurs tactiles qui rendent tout autre circuit obsolète. Cela dit, le circuit d'entrée est seulement le tactile lui-même et non l'afficheur, parce que si l'utilisateur est le concepteur du système lui-même en suivant exactement le déroulement du processus environnemental qui affecte ce système même si l'afficheur ne fonctionne pas, si le tactile fonctionne le déroulement du fonctionnement du système continu le plus normalement.

Tous ces caractéristiques que nous venons de mentionner n'ont surgis que pour une meilleure commercialisation du système embarqué conçu.

I. 8) Conclusion

Lorsqu'un système est utilisé pour une tâche bien précise, il est souvent plus efficace et économe s'il est spécifique à cette fonctionnalité que s'il est général. Les systèmes embarqués sont très souvent utilisés dans ces conditions, et il est donc intéressant qu'ils soient conçus spécifiquement pour les fonctions qu'ils doivent remplir.

Chapitre 2

Généralités sur les systèmes d'alarmes d'intrusion et de contrôle d'accès

I. 1) Introduction

La sécurité revêt une importance primordiale pour tout édifice. Cela a conduit à l'apparition des systèmes de sécurité. On distingue plusieurs types, par exemple : les systèmes de surveillance, les systèmes de contrôles d'accès et les systèmes de protection contre les incendies. L'ensemble de ces systèmes de sécurité assure une protection à la fois innovante et efficace pour son utilisateur.

Dans ce travail on traite le problème de conception et de réalisation d'un système d'alarme contre les intrusions. Ce système permettra de détecter toute présence d'intrus dans un local et transmettra instantanément cette information au propriétaire et ce, quel que soit sa localisation grâce à l'envoi de SMS via le réseau GSM.

Ces systèmes de détection sont largement utilisés dans le secteur industriel personnel. Ces dispositifs permettent de sécuriser les entreprises et les domiciles et sont même utilisés comme moyen de surveillance (ex : Parking, local de stockage…). Actuellement on assiste à une généralisation de l'utilisation de ces systèmes de sécurité. En effet leur prix devient abordable, et leur installation est de plus en plus facile.

Un système anti-intrusion ou du moins para-intrusion englobe tout un dispositif électronique et informatique pour répondre à ses besoins.

I. 2) Système de détection des intrusions

Les dispositifs de sécurité assurent la protection en trois étapes:

- détection
- retardement
- alarme

Les systèmes d'alarme industriels résultent d'une intégration de plusieurs systèmes de capteurs. Ces derniers sont placés dans une barrière extérieure (à l'extérieur de l'édifice et à l'intérieur de la barrière de protection). Cet ensemble de capteurs permet ainsi de détecter et de retarder l'intrus avant même qu'il n'atteigne le local.

Dans les foyers, la tâche est beaucoup plus complexe. La variation de l'architecture des foyers qui sont probablement bâtis sans avoir prévu l'installation d'un système de sécurité, complique d'avantage le choix de l'implémentation des capteurs. Ajouté à cela le voisinage et l'environnement,

Si les capteurs sont montés sur des barrières, disposés aux dessus d'un mur ou enfouis dans le sol, permettent au système de sécurité de détecter l'intrus et de le retarder (simulation de présence, ou libération de la cage à chiens), l'implémentation de ces derniers a l'intérieure d'un édifice rend obsolète l'idée de retardement.

Au vu de ces contraintes, on comprend que le choix de l'emplacement des capteurs est crucial.

I. 3) Contrôle d'accès

En matière de sécurité physique, le terme contrôle d'accès ou contrôle de parenté désigne le fait de restreindre l'entrée d'une propriété aux personnes autorisées. Le contrôle d'accès physique peut être réalisé par un gardien, par des moyens mécaniques tels que des serrures à clés, ou par des moyens technologiques tels que les systèmes automatiques de contrôle d'accès.

Un système de contrôle d'accès détermine qui est autorisé à entrer ou sortir, et/ou quand ils sont autorisés à entrer ou sortir.

Historiquement, cela a été partiellement réalisé au moyen de clés et des serrures. Quoique les serrures mécaniques à clés ne permettent pas des restrictions temporelles d'accès tout en étant obligé de revoir les verrous quand une clé mécanique est perdue ou le détenteur de la clef n'est plus autorisé à utiliser la zone protégée avec tous le risque d'une copie de la clef.

Le contrôle d'accès électronique utilise des systèmes plus intelligents pour éviter ces écueils, en offrant un large éventail de moyens permettant de remplacer les dispositifs classiques.

Le contrôle électronique accorde l'accès en se basant sur les informations d'identification présentées. Lorsque l'accès est accordé, la porte est déverrouillée et la transaction peut être comptabilisée. Lorsque l'accès est refusé, la porte reste verrouillée et la tentative d'accès est enregistrée. Le système peut également surveiller la porte et déclencher une alarme si la porte est forcée ou maintenue ouverte trop longtemps après avoir été déverrouillée [12].

Cependant il y a des systèmes de contrôle d'accès et qui jouent le rôle de contrôles de parenté. On peut même parler de vérification d'accès, en se positionnant bien à l'intérieure de la propriété, proposant un temps défini après une éventuelle intrusion, temps pendant lequel le système reste à l'écoute du contrôle de parenté ; après cela l'alarme est enclenchée.

I. 4) Types de capteurs d'intrusion à l'intérieur [10]

a. Capteurs à infrarouge passif

Les capteurs à infrarouge passifs (Passive Infra Red: PIR) sont les capteurs les plus utilisés dans les environnements domestiques et les petites entreprises, car ils offrent des fonctionnalités fiables et abordables. Le terme "passif" signifie que le détecteur est capable de fonctionner sans avoir besoin de générer et émettre sa propre énergie (contrairement aux capteurs à ultrasons et à micro-ondes qui sont des détecteurs d'intrusion volumétriques "actifs"). Les PIR sont capables de distinguer si un objet émetteur infrarouge est présent d'abord par la détection de la température ambiante de l'espace surveillé, puis par la détection d'un changement dans la température causée par la présence de cet objet. En utilisant le principe de différenciation, qui se traduit par une vérification de la présence ou non-présence.

b. Détecteurs à ultrasons

Utilisant des fréquences entre 25 kHz et 75 kHz, les détecteurs à ultrasons actifs émettent des ondes sonores inaudibles par l'être humain. Ces ondes sont réfléchies par des objets solides (tels que le sol, le mur et le plafond), puis captées par le récepteur du capteur. Ils permettent ainsi de détecter une éventuelle présence d'un être vivant grâce à l'effet Doppler. Le principe est que les ondes ultrasonores sont presque complètement réfléchies par les objets à surface rigide alors que les objets à surface molle (comme le corps humain) ont tendance à absorber une partie de l'énergie de ces ondes et entraînent ainsi un changement de leur fréquence.

c. Détecteur à micro-ondes

De même principe que le précédent, ce détecteur émet des micro-ondes, et les détecte après qu'elles soient réfléchies tout en mesurant leur intensité.

d. Détecteur à faisceaux lumineux modulés

Le système à faisceau photoélectrique détecte la présence d'un intrus par l'émission visible ou infrarouge des faisceaux lumineux dans une zone. Pour améliorer la surface de détection, les faisceaux sont souvent employés en double capteurs ou plus. Cela permet d'empêcher le contournement par l'intrus, puisque la surface couverte par un capteur est un simple rayonnement.

Si ce type de systèmes est valable pour des applications internes, il est tout aussi intéressant pour les applications externes en faisant pointer le capteur de manière à effleurer la porte.

e. Détecteurs de bris de vitre

Le détecteur de bris de verre peut être utilisé pour la protection du périmètre de construction interne. Quand le verre se casse il génère du son dans une large bande de fréquences (couvrant les fréquences sonores audibles et ultrasonores inaudibles, allant de quelques Hz à plus de 20 kHz). Les détecteurs de bris de verre acoustiques sont montés à proximité des vitres et ils surveillent les fréquences sonores causées par le bris du verre.

I. 5) Types de capteurs d'intrusion à l'extérieur (En plein air) [11]

Ces types de capteurs se trouvent souvent montés, sur des barrières ou installés sur le périmètre de la zone protégée.

a. Vibreur

Ces dispositifs sont montés sur des obstacles et sont surtout utilisés pour détecter une attaque sur la structure elle-même. L'idée exploite une configuration mécanique instable qui fait partie du circuit électrique. Alors quand un mouvement ou une vibration se produit, la partie instable du circuit se déplace et brise le flux de courant, pour produire un signal d'alarme.

L'avantage de ces capteurs est qu'ils sont très fiables, à faible taux de fausses alarmes avec un prix abordable. Leurs inconvénients est qu'ils doivent être montés sur la clôture. Le prix de l'implantation dépasse le prix d'achat.

b. Détection passive du champ magnétique

Ce système de sécurité, est basé sur le principe de détection des anomalies magnétiques de l'opération d'intrusion. Le système utilise un générateur de champ électromagnétique alimenté par deux câbles en parallèle. Les deux fils passent le long du périmètre et sont généralement installés à environ 5 cm au-dessus d'un mur ou d'environ 30 cm sous terre. Les fils sont connectés à un processeur de signal qui analyse tout changement dans le champ magnétique.

L'avantage de ces capteurs est leurs faible taux de fausse alarme. Cela dit ils ne peuvent être installé à proximité de lignes à haute tension, les radars ou les aéroports.

c. Détection active du champ électromagnétique

Ce système de proximité peut être installé sur le périmètre du bâtiment, des clôtures et des murs. Il offre aussi la possibilité d'être installé sur des poteaux autoportants dédiés. Le système utilise un générateur de champ électromagnétique alimentant un fil, avec un autre fil de détection parallèle. Les deux fils passent le long du périmètre et sont généralement

installés, près de 800mm, l'un par rapport à l'autre. Le fil de la sonde est relié à un processeur de signal qui analyse:

- la variation d'amplitude du champ
- le changement de taux électromagnétique (mouvement des intrus)
- le temps des perturbations

Ces paramètres caractérisent le mouvement de l'intrus et quand les trois sont détectés simultanément, un signal d'alarme est généré. La barrière peut fournir une protection contre le sol à environ 4 mètres d'altitude. Il est généralement configuré dans les zones de longueurs allant jusqu'à 200 mètres selon le nombre de fils du capteur installé.

On cite comme avantage leur complète discrétion. Quoique leurs coûts élevés et leurs fiabilités jouent en leurs défaveurs, ajouté à cela leurs dépendance des conditions météorologiques.

d. Clôture à micro-ondes

Ce type de dispositifs produit un faisceau électromagnétique utilisant des ondes à hautes fréquences qui passent de l'émetteur au récepteur, ce qui crée un mur invisible, mais sensible à l'intrusion. Lorsque le récepteur détecte un changement dans le faisceau reçu (dû à une éventuelle intrusion), le système démarre une analyse détaillée de la situation et peut déclencher, le cas échéant, un signal d'alarme.

Parmi leurs avantages on distingue, l'invisibilité (barrière invisible), le faible coût et la facilité de mise en œuvre. Leurs inconvénients se réfèrent à la nécessité d'être implantés dans des espaces dégagés, et la sensibilité aux conditions météorologique.

e. Détecteur à fibre optique

Un détecteur à fibre optique peut être utilisé pour détecter les intrusions en mesurant la différence de la quantité de lumière envoyée par le noyau de la fibre. Si la fibre est perturbée, une partie de la lumière sera perdue et le récepteur détecte cette fuite. La détection peut porter aussi, non pas sur la quantité de la lumière reçue, mais sur le changement de polarisation causé par le mouvement survenu sur la fibre.

Le support portant la fibre peut être attaché directement à une clôture ou lié à une bande en acier barbelé qui est utilisé pour protéger le haut des murs et des clôtures.

La faculté de couverture des longues distances constitue un avantage. Par contre le taux élevé de fausses alarmes n'est pas négligeable.

I. 6) Type de signalisation

La signalisation a pour but d'informer sur une éventuelle intrusion ; Nous distinguons trois types de signalisation : préventive, dissuasive ou curative [13X].

Signalisation préventive

La signalisation préventive fait référence à tout type de projecteur. Son but est de mettre l'intrus dans le doute dès qu'il y a franchissement de la zone périphérique.

Signalisation dissuasive

Cette signalisation est représentée par tout dispositif sirène pouvant être à l'intérieure comme à l'extérieure d'une propriété. Elle a pour but, l'avertissement du voisinage (à l'intérieure et a l'extérieure), ou l'agression du système auditif de l'intrus par une émission sonore au-delà du seuil de la douleur (à l'intérieure).

Signalisation curative

La signalisation curative fait référence à tout type de transmission à distance. Elle a pour but la signalisation de l'intrus à une tierce personne afin de mettre en œuvre une intervention. Parmi les signalisations curatives, nous mentionnons, le transmetteur téléphonique, SMS, la transmission vidéo (Capture vidéo), …etc.

I. 7) Conclusion

Dans ce chapitre nous avons détaillé les systèmes d'intrusions par leurs types, leurs structures, et leurs périphériques.

Nous avons aussi cité quelques avantages et inconvénients de ces différents périphériques. De même nous avons donné une vue globale sur les systèmes de contrôle d'accès, leurs utilisations ainsi que leurs rôles. Nous avons terminé par la citation des différents types de signalisation de l'intrus.

Chapitre 3

Conception du système de surveillance et contrôle d'accès

III. 1) Introduction

Nous proposons dans ce chapitre le procédé de mise en place de notre système de détection d'intrusion afin de protéger les locaux industriels et personnels. Ce système sera équipé d'un moyen de communication (envoi de SMS via le réseau mobile) permettant de transmettre une signalisation d'alarme à ceux qui doivent intervenir en cas d'incident, et d'une sirène.

Notre système doit être muni de capteurs pour la détection d'intrus, d'un mécanisme de contrôle d'accès (authentification par mot de passe) pour la porte principale.

III. 2) Fonctionnalités du système

Le système doit assurer les fonctionnalités suivantes :

- surveillance des locaux, afin d'assurer un certain degré de sécurité
  - ✓ 4 zones de surveillance
  - ✓ Possibilités d'utiliser des détecteurs universels. (capteur magnétique, infrarouge, ILS…)
  - ✓ Possibilité d'extension
- Authentification après détection
  - ✓ Mot de passe secret
- Signalisation en cas de non authentification
  - ✓ Envoie d'un SMS
  - ✓ Déclenchement d'une sirène

Du point de vue technique, le système doit être :

- ✓ Autonome (indépendant de tout autre système)
- ✓ Facilement configurable à travers un clavier et un écran LCD offrant à la fois la simplicité de la configuration et la visualisation des informations que le système doit fournir à l'administrateur.
- ✓ Fiable
- ✓ d'un coût relativement abordable par rapport aux systèmes disponibles sur le marché.

Pour répondre à ces exigences, le système de surveillance sera composé de plusieurs modules. (Module de contrôle, module de détection, interface GSM, interface utilisateur)

III. 3) Schéma synoptique du système

Le système de surveillance est constitué de plusieurs modules représentés dans la figure 2.

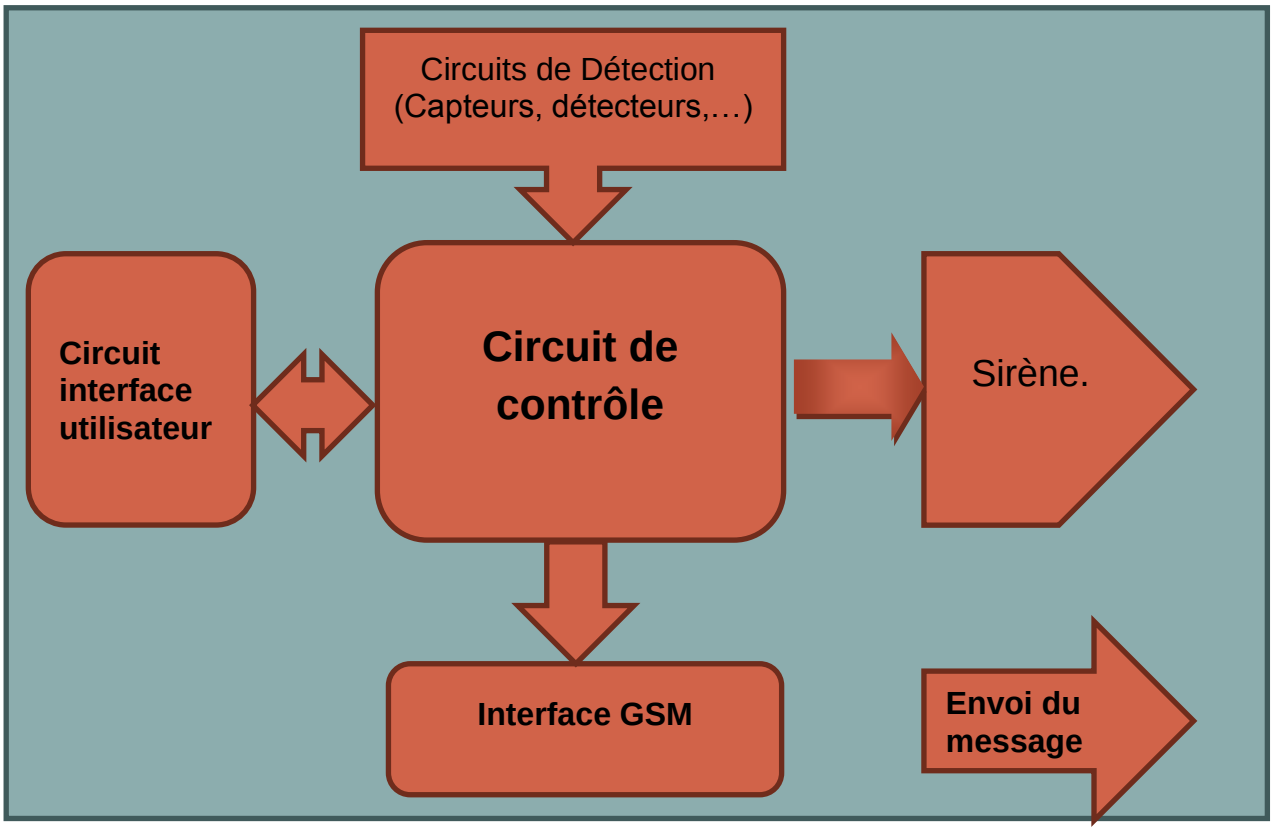


Figure 2 Schéma bloc de système

- le circuit de contrôle prend en charge la collecte, le traitement et la transmission des données. C'est lui qui gère et qui assure le fonctionnement du système.
- Le circuit interface utilisateur assure le paramétrage à travers un clavier, et informe sur l'état de système via l'afficheur LCD.

- L'interface GSM: assure le transfert d'information concernant l'état de l'alarme en cas d'intrusion via le SMS.

III. 3 .1). Principaux éléments constituant chaque module

- **Module contrôle**

Il constitue le cœur du système. Il est maître des autres modules desquels il reçoit les informations et vers lesquels il envoie ses commandes.

Ce module accomplit les tâches suivantes :

- ✓ Gestion de l'authentification (reconnaissance et changement de mots de passes)
- ✓ basculement entre les états du système.
- ✓ Configuration des temporisations
- ✓ Déclenchement de l'alarme, envoie de SMS.

- **Module interface utilisateur**

Il est constitué d'un clavier et d'un écran LCD pour interagir avec l'utilisateur, il assure la saisie des différents paramètres du système.

- **Interface GSM**

Suite au déclenchement d'alarme (tentative d'intrusion), ce module reçoit une demande de la part du module central pour établir un contact téléphonique avec le propriétaire prédéfini, pour lui signaler l'intrusion à travers un SMS.

III. 3 .2). Critères de choix des composants

Les composants choisis pour réaliser ce projet sont les suivants :

- un microcontrôleur Atmel basé sur un processeur ARM7TDMI pour le module central.
- Un écran à cristaux liquides (LCD 2*16).
- Une interface GSM de type M2M (machine to machine) (module TM2).
- Un clavier matriciel hexadécimal
- Des détecteurs d'intrusion

a) Le choix du microcontrôleur

- Nombre d'entrées sorties nécessaires à notre système:
  - un clavier à 16 touches qui nécessite 8 entrées/sorties

- une interface USART à deux entrées, nécessaire pour le module TM2.
- des capteurs (au moins 4 entrées, une par zone).
- un afficheur LCD qui nécessite 6 entrées/sorties,

Ainsi, la totalité des entrées/sorties nécessaires s'élève à 20. D'où la nécessité d'utiliser un microcontrôleur supportant au moins 20 entrées/sorties. Le microcontrôleur utilisé dispose de 32 broches pouvant être utilisées comme entrées/sorties

- Taille de la RAM suffisamment large pour éviter d'avoir recours à des mémoires externes. Le microcontrôleur dispose de 256Ko de RAM largement suffisant pour notre système
- Puissance de calcul suffisante pour une application temps réel

b) Le clavier hexadécimal

Ce clavier suffit pour faire saisir un code, pour activer ou désactiver le système de surveillance, ainsi que pour faire un changement de la configuration qui est proposé.

c) L'écran LCD

Pour pouvoir communiquer les informations sur le système à l'utilisateur, en l'assistant dans la saisie des différents paramètres (numéro de tél à utiliser, temporisation, …etc)

d) L'interface GSM

Le choix est fait par la disposition d'une interface machine-machine (M&M) offrant la possibilité de transmettre et recevoir des messages SMS.

III. 4) Description fonctionnelle des composants

Après avoir fait le choix sur les composants a utilisé nous allons détailler la description fonctionnelle de ces derniers.

III. 4 . 1) Description de la carte Atmel AT91SAM7

A. Le processeur ARM7TDMI

Différent des autres composants constituant notre système, le microprocesseur ARM7TDMI représente le cerveau du système lui-même. Proposé par l'entreprise ARM, cette dernière ne fait que breveter des architectures de processeurs. La conception de cette architecture est réalisée par toute entreprise qui relève le brevet.

Ayant été développé par l'équipe anglaise Acorn Computer à Cambridge et donné comme acronyme Acorn Risc Machine, la société ARM relève le flambeau et donne Advanced Risc Machine comme acronyme de l'ARM [14]. Les processeurs basés sur les cœurs ARM ont envahi notre quotidien avec déjà plus de 5 milliards d'unités vendues, proposant une large gamme de processeurs adaptés aux applications embarquées gourmandes en ressources de calcul par un rapport performance/consommation imbattable. Nous les retrouvons notamment dans des appareils aussi variés que des téléphones mobiles, des imprimantes, des équipements réseau, des appareils multimédias, des consoles de jeu, des appareils photos, des ordinateurs de voiture, etc...

Les cœurs de processeurs ARM partagent une architecture commune qui peut être résumée ainsi :

– Processeurs RISC (Reduced Instruction Set Computer) :

- File de registres uniformes indifférenciés.
- Architecture « load/store » : seules les instructions de chargement et rangement accèdent à la mémoire ; toutes les autres instructions (traitements) travaillent sur les registres.
- Modes d'adressages réduits, opérant uniquement à partir des valeurs des registres et de décalages.
- Instructions de longueur fixe et de structure uniforme afin d'accélérer le décodage.

– Possibilité d'utiliser l'unité arithmétique et logique pour chaque instruction de traitement.

– Gestion des modes d'adressage auto-incrémentés et auto-décrémentés.

– Gestion des instructions de chargement et rangement par blocs.

– Exécution conditionnelle de toutes les instructions.

Ces caractéristiques garantissent des performances maximales, eu égard à la consommation et à la complexité du processeur [15].

Le cœur de processeurs ARM7TDMI a fait naissance dans la version 4, avec un ensemble de caractéristiques résumées comme suit [16]:

L'arm7TDMI (Thumb, débuggin, fast multiplier, In circuit Emulator), propose 31 registres généraux de 32 bits et six registres d'état qui cependant, à chaque instant, seuls 16 registres généraux et deux registres d'état sont visibles ; les autres sont masqués selon le mode de fonctionnement du processeur et qui peuvent être déclenchés sur instruction ou sur évènement. L'ARM dispose de sept modes de fonctionnement différents, mentionnés ci-après :

– USR (USeR mode) : mode utilisateur. Il s'agit du mode de fonctionnement des programmes utilisateurs. Certaines ressources sont inaccessibles (il s'agit du seul mode de fonctionnement non privilégié), et il est impossible de changer de mode autrement qu'en causant une interruption logicielle.

– IRQ (Interrupt ReQuest mode) : mode interruption (les registres R13 et R14 sont remplacés par des copies privées). Ce mode est utilisé pour gérer les interruptions matérielles externes : il devient actif lorsqu'un signal actif est reçu sur l'entrée IRQ du processeur.

– FIQ (Fast Interrupt reQuest mode) : mode interruption rapide (les registres R8 à R14 sont remplacés par des copies privées1). Ce mode est actif lors de la réception d'un signal sur l'entrée FIQ du processeur. Ce mode se distingue du précédent par des capacités supérieures en terme de transfert de données, et un nombre de registres dédie supérieur. La priorité du mode FIQ est supérieure à celle du mode IRQ.

– SVC (Supervisor mode) : mode Superviseur (les registres R13 et R14 sont remplacés par des copies privées). C'est un mode protégé, utilisé pour l'exécution du système d'exploitation.

– Abort : signalé par le gestionnaire de la mémoire, utilisé pour implémenter des mécanismes de mémoire virtuelle et/ou de protection mémoire.

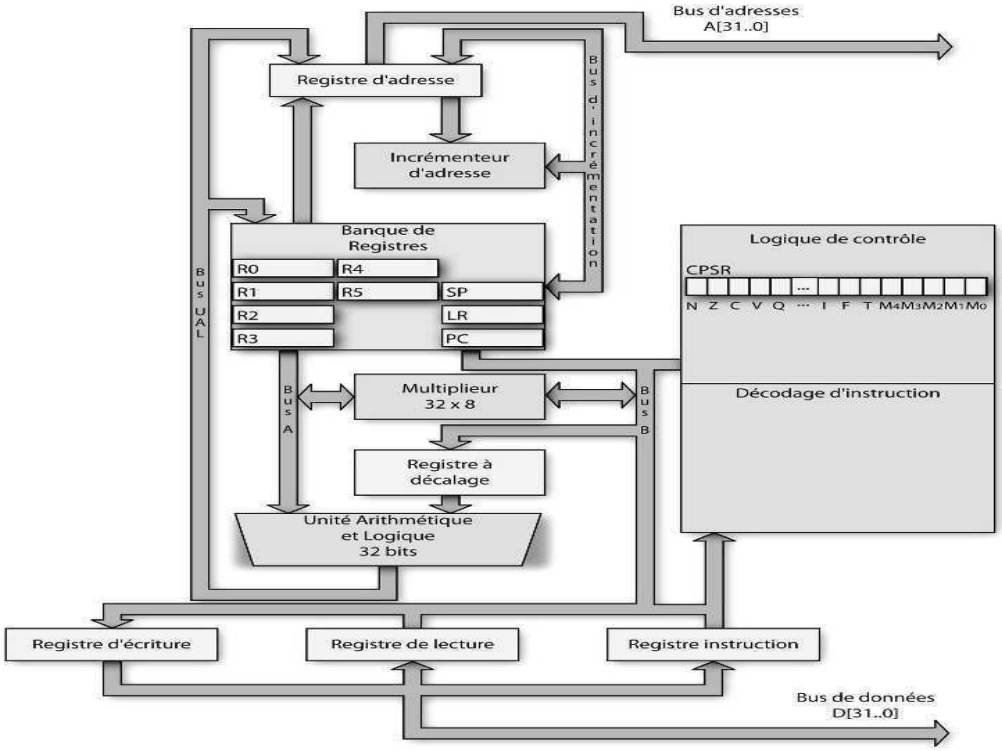
– Undefined : utilisé pour l'émulation logicielle de coprocesseur.

– System (sys) : mode utilisé par le système d'exploitation pour effectuer des tâches nécessitant des privilèges étendus. Ce mode est donc similaire au mode supervisor, à ceci près que les registres sont ceux accessibles en mode utilisateur.

Proposition du choix de gestion de la mémoire (little-endian ou alors bigendian). Une mémoire cache à 8 KO.

B.1) Architecture interne de l'ARM7TDMI

Le détail de la structure interne du cœur de processeur ARM7 est représenté sur la figure 3 :



.

Figure 3 Architecture d'un processeur arm

L'une des premières choses à mentionner dans ce processeur est que son architecture est de type van Neumann ; seulement deux bus sont utilisés pour véhiculer l'information, l'adresse sur le bus d'adresse, et la donnée sur le bus de donnée ; de ce fait on peut facilement deviner qu'ils mènent à la même zone mémoire. Il n'existe pas de séparation entre zone mémoire de données ou de programmation. Présentant un pipeline à 3 étages (lecture d'instruction, décodage d'instruction et exécution d'instruction), à chaque cycle où une lecture d'instruction est possible, l'adresse présente dans le registre d'adresse est donc déposée sur le bus d'adresse pour obtenir de la mémoire l'instruction suivant celle qui est en

cours de décodage. Notons qu'un incrémentateur d'adresse permet d'obtenir directement l'adresse de l'instruction suivante sans solliciter l'unité arithmétique et logique. L'instruction ainsi obtenue de la mémoire est alors écrite dans le registre instruction; pour être transférée dans la logique de décodage lors du cycle suivant. La logique de contrôle enfin, est en charge de l'exécution proprement dite de l'instruction. Elle sélectionne l'opération à effectuer, le registre opérande source, ainsi que les registre opérande destination. Elle actionne pour cela les barrières trois états des registres sources, et la commande d'écriture du registre destination. Notons qu'une des entrées de l'unité arithmétique et logique passe au travers d'un circuit décaleur combinatoire (ce qui permet de faire des décalages et une opération arithmétique en une seule instruction).

B. Les Entrées sorties implémentées par la carte Atmel

Dans cette section nous portons une brève attention sur les fonctionnalités de dialogue entre le processeur de la carte Atmel et le monde extérieur.

Notons que dans le processeur ARM7, l'espace d'adressage dédié à la mémoire et le même que l'espace d'adressage dédié aux entrées/sorties. La technique consiste à faire correspondre dans la mémoire classique, des zones indexées par les circuits d'entrées/sortie (le Mapping mémoire), et les opérations utilisées pour lire et écrire les valeurs des registres de données, d'état et de contrôle sont les mêmes que les instructions utilisées pour lire et écrire la mémoire.

L'avantage de cette stratégie réside dans une simplicité accrue de la structure interne du processeur (les instructions spécifiques d'entrées/sorties ne sont plus nécessaires). Il est néanmoins de la responsabilité du programmeur de bien veiller à savoir, à chaque lecture/écriture, s'il accède à de la mémoire ou à un circuit d'entrées/sorties.

Le système propose deux techniques de dialogue avec les périphériques, et qui sont les entrées/sorties par teste d'état dites aussi par scrutation, et les entrées/sorties par interruption. Dans les entrées/sorties par teste d'état, tout circuits d'entrée/sortie comporte un drapeau (une bascule) dans son registre d'état qui représente un (ou plusieurs, selon la complexité du périphérique) bit(s) positionné(s) à 0 ou 1. Pour traiter une entrées/sorties conditionnée à l'état d'un périphérique, le processeur boucle sur la lecture du registre d'état correspondant jusqu'à satisfaction de la condition. On procède donc à une attente active [15].

L’inconvénient majeur des entrées/sorties par test d’état réside dans le fait que le dialogue entre les circuits et le processeur étant synchrone, ce dernier reste sollicité pendant toute la durée de la transaction; cet écueil est contourné dans les entrées/sorties dites par interruption. Le principe est de mettre en place un mécanisme grâce auquel un circuit d’entrées/sortie va pouvoir signaler au processeur que son attention est requise. C’est le principe des interruptions.

C. Architecteur interne de la carte Atmel

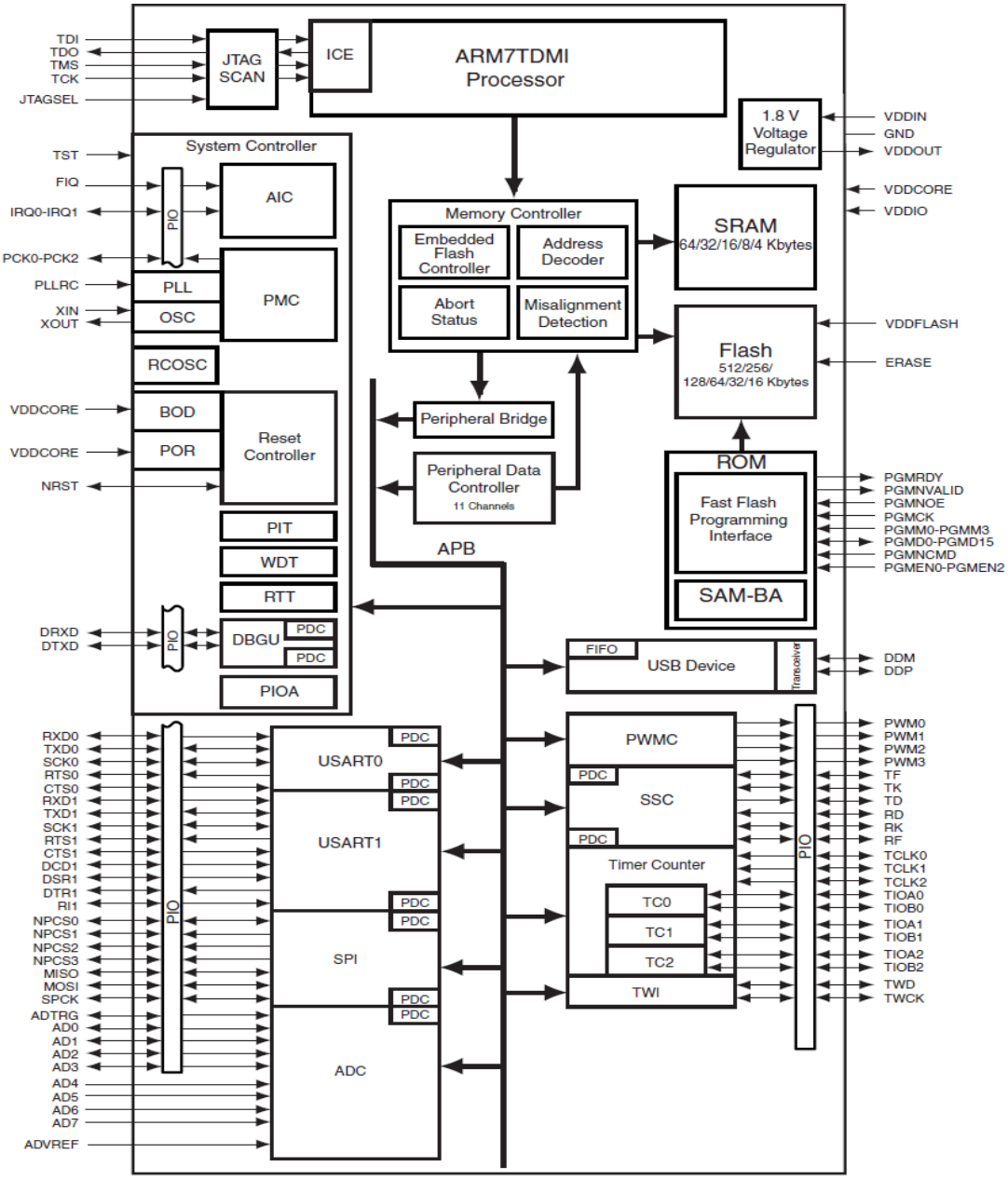


Figure 4 Architecteur interne de la carte Atmel

D. Les unités fonctionnelles de base :

Le bloc fonctionnel nommé « System Controller » contient tous les éléments de base nécessaires au bon fonctionnement du processeur et qui sont résumés comme suit [17]:

a) Le PMC (Power Management Controller)

Ayant pour but de minimiser la consommation d'énergie, il contrôle l'ensemble des horloges système [17]

- MCK (Master Clock) alimente les périphériques qui fonctionnent en permanence (AIC et MC).
- PCK (Processor Clock) alimente le processeur (arrêté lorsque le processeur est en mode oisif « idle »).
- UDP Clock (USB Device Port Clock) alimente le port USB.
- Les horloges destinées aux périphériques (USART, SPI, etc.) peuvent être stoppées pour désactiver de manière sélective ces périphériques et ainsi économiser la consommation de l'énergie.
- Des horloges en sorties, disponibles sur les broches PCK.

b) Le RC (ou RSTC, Reset Controller)

Gère les lignes reset de l'AT91 : le reset du processeur, le reset des périphériques, ainsi que la gestion de la ligne nRST. Il donne l'état de la dernière remise à zéro, indiquant s'il s'agit d'une réinitialisation du logiciel, de l'utilisateur, du chien de garde ou de mise sous tension [17]

c) Le PIT (Periodic Interval Timer)

Destiné à gérer le temps partagé au niveau du système d'exploitation. Le PIT est arrêté lorsque le noyau passe à l'état de débogage [17].

d) Le RTT (Real Time Timer)

Typiquement utilisé pour battre l'unité de temps qu'est la seconde. Il utilise une base de temps de faible acuité (slow clock à 32,768 kHz). Il est construit autour d'un compteur 32 bits(Le compteur 32 bits peut compter jusqu'à 2^{32} secondes, ce qui correspond à plus de 136 ans et alors réinitialisé) et permet de compter les secondes écoulées. Il génère une interruption périodique et / ou déclenche une alarme sur une valeur programmée [17].

e) Le DBGU (Debug Unit)

Gère tout l'aspect débogage du SAM7S

f) La PIO (Parallel Input Output)

Chargé de gérer les entrées/sorties. Constitué de 32 broches. La figure 5 illustre la structure d'un port de chaque broche de la PIO:

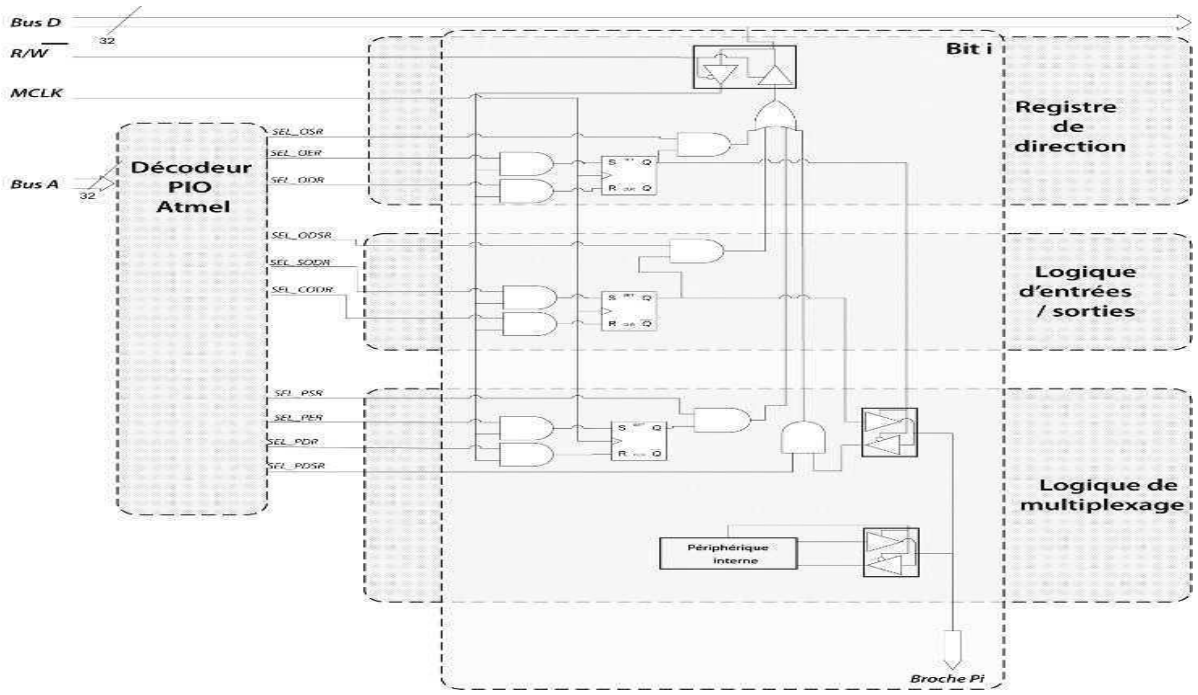


Figure 5 Architecture interne de la PIO

- **Description fonctionnelle**

Nous distinguons trois bascules sur notre schéma. Le « registre de direction » permet de déterminer le sens du dialogue avec le processeur, soit en entrée ou en sortie. Le registre « logique d'entrée/sortie » qui, en fonction de la direction, issue du registre vu précédemment, la valeur produite sur le bus D est envoyée vers le milieu extérieur (port en sortie) ou inversement (port en entrée). Le troisième registre (« logique de multiplexage ») permet de déterminer si la broche, dans le cas où elle est multiplexée, sert au port parallèle ou à un autre périphérique. D'autant plus qu'une application n'a pas souvent recours à l'ensemble des ressources cités plus haut, les broches non utilisées seront disponibles pour l'exploitation des E/S dites parallèles.

Les broches s'élèvent aux nombre de 6 continuellement disposées aux E/S parallèles (outre quatre broches d'entrée d'interruption.), et peuvent s'accroitre jusqu'à atteindre 32 broches selon l'exploitation des circuits interne (un seul circuit peut être multiplexé avec plus d'une broche).

Dans la logique de décodage chaque bloc fonctionnel (registre de direction, logique d'entrée/ sortie, logique de multiplexage) est architecturé autour d'une bascule RS pilotée par deux signaux (l'un se terminant par « ER », l'autre par « DR ») et dont la valeur est donnée par un troisième signal (terminé par « SR »). À l'échelle du PIO (et pour les 32 bits donc), ces bascules forment autant de registres dont le fonctionnement est le suivant.

Le bloc SR (Status Register) correspond à un registre formé de bascules RS. L'entrée S de chaque bascule est reliée au bit correspondant du bloc ER (Enable Register), l'entrée R au bit correspondant du bloc DR (Disable Register). Ainsi donc, à chaque triplet ne correspond en fait qu'un seul registre – celui dans lequel est stocké l'état.

g) AIC : (Advanced Interrupt Controller) :

Le contrôleur d'interruptions AIC est constitué de deux entrées pour les interruptions, qui sont FIQ et IRQ, la gestion de la priorité et de la vectorisation des interruptions pour le processeur, nécessite la présence d'un circuit dédié pour ça, avec le schéma fonctionnel suivant [17]

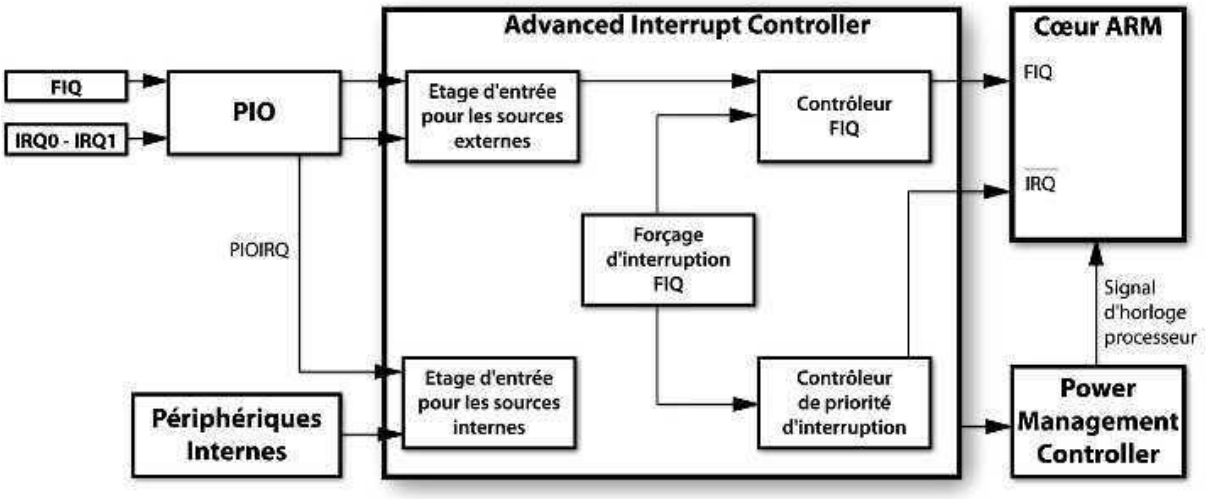


Figure 6 Diagramme de bloc de l'AIC

L'AIC est capable de gérer jusqu'à 32 sources d'interruptions possibles, tout en proposant huit niveaux de priorité.

h) MC : (memory controller)

Le contrôleur mémoire gère le bus interne ASB et arbitre les accès à la mémoire des deux masters (processeur arm7tdmi ou le contrôleur de périphérique DMA). It est composé de [17]:

-Un arbitre de bus.

-Un décodeur d'adresse.

-Un état d'abandon.

-Détecteur de défaut d'alignement.

- Un contrôleur de mémoire flash embarqué.

Le MC ne gère que le mode d'accès LITTLE-ENDIAN, du fait que les deux masters sont en ce même mode.

- **Arbitre de bus**

C'est un simple bus avec un dispositif conçu pour donner la plus grande priorité au périphérique DMA. L'ARM7TDMI a une priorité seconde.

- **Décodeur d'adresse**

Le décodeur d'adresse dispose d'un décodeur pour distinguer trois (3) zones définies comme suit :

- un espace d'adressage de 256 Mo dédié pour les mémoires internes.
- un espace d'adressage de 256 Mo réservé pour les périphériques embarqués.
- Un espace d'adressage indéfini de 3584 Mo représentant quatorze zones de 256 M octets qui renvoient un abort si on y accède.

- **Un contrôleur de flash embarqué**

Il est embarqué dans la MC, il assure l'interface du bloc flash avec le bus interne. Il augmente la performance pour le mode thumb, il gère aussi la programmation, l'effacement, le verrouillage et le déverrouillage des séquences grâce à un ensemble complet de commandes.

- **Détecteur de défaut d'alignement :**

Ayant pour rôle de vérifier la cohérence des accès, le dispositif pour chaque tentative d'accès, compare la taille du mot avec les deux premiers bits de son bus d'adresse.

Si le type d'accès est un mot (32 bits) et les deux premiers bits ne sont pas à zéro, ou si le type de l'accès est un demi-mot (16 bits) et le premier bit n'est pas à zéro, un arrêt prématuré est renvoyé pour le maître et l'accès est annulé.

- **Abort status**

Abort status est intégré dans la MC, et ce principalement afin de facilité de débogage et/ou l'analyse des erreurs par le système opératoire.

Il y a trois raisons pour qu'un abort status soit sollicité :

- ✓ l'accès à une adresse non définie
- ✓ un accès à une adresse non alignée.
- ✓ une interruption se produit.

i) TC (timer contner) :

Le timer contner est utilisé pour effectuer des mesures de fréquences ou d'intervalles de temps, du comptage d'événements, de la génération de signal, etc. contenant trois timers identiques dont chacun s'appuie sur un compteur 16 bits, automatiquement incrémenté par le biais d'une horloge qui peut être [17]:

- ➢ Interne : il s'agit alors de l'horloge principale (Master Clock) divisée par 2, 8, 32,128 ou 1 024.
- ➢ Externe : on a, dans ce cas, le choix entre trois signaux différents paramétrables.

Chaque timer peut être paramétré et utilisé indépendamment des autres.

La figure 7 représente Diagramme de bloc des timers de l'AT91.

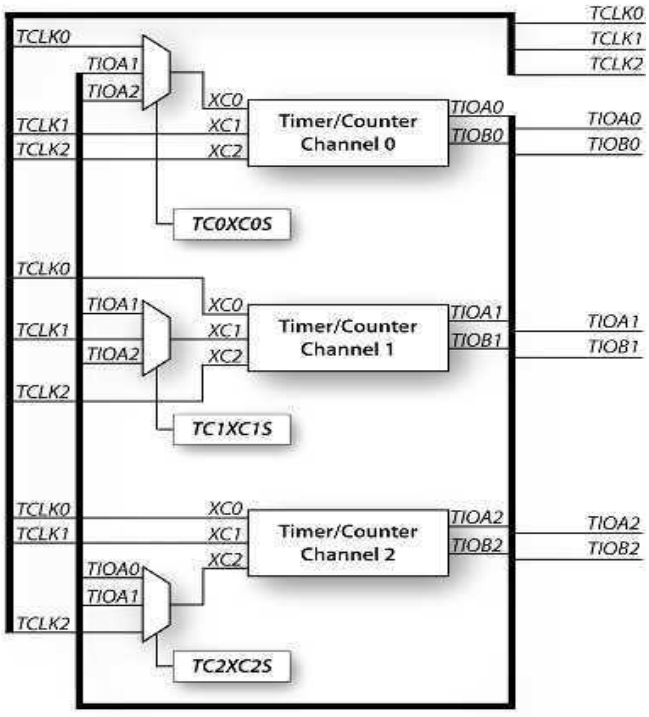


Figure 7 Diagramme de bloc des Timers de l'AT91

III. 4 . 2) Présentation du module réseau TM2

a) MODULES GSM INTÉGRÉS

Les modules GSM intégrés sont des modules débarrassés de leurs interfaces homme-machine pour qu'il ne subsiste que la partie interface machine-machine (M2M), qui physiquement correspond à un connecteur multibroche quelconque ou encore à un connecteur DB9 facilitant la connexion à un PC ou dans notre cas à un microcontrôleur. Ces modules sont universels puisqu'ils supportent les normes GSM07.07 et GSM07.05 permettent de ce fait l'échange de données, de SMS, d'emails et même de télécopies (FAX) via le réseau de téléphonie mobile. Leurs simplicités de mise en œuvre ouvrent des perspectives très intéressantes concernant la réalisation de montages électroniques sans fil. [18]

b) Le TM2 de TELTONIKA

Parmi les différents modèles proposés par la société Lextronic (constructeur de modules GSM intégrés) , on retrouve le module TM2 qui est un modèle quadri bandes qui utilise les fréquences 850, 900, 1 800 et 1 900 MHz. Il est capable de fonctionner dans les modes voix, données, FAX et surtout, le plus intéressant pour nous, dans le mode SMS. Le module dispose d'un support destiné à recevoir une carte SIM et un connecteur MMCX permettant de relier une petite antenne RF (radio fréquence) également fournie par Lextronic.

Toutes les entrées et sorties utiles au pilotage du module sont disponibles sur un connecteur comportant 60 points en CMS. Lextronic propose un adaptateur qui répartit l'ensemble des connexions sur 4 rangées de 15 points au pas classique de 2,54mm.

Le tableau 1 montre ses caractéristiques principales

Transmission	Voix, données et SMS
Alimentation	3.5 V à 4.2 V, typiquement 3.8 V
Bandes de fréquences	GSM 850 MHz, EGSM 900 MHz, DCS 1800 MHz, PCS 1900 MHz
Courant absorbé	GSM900 : 147 mA (900 mA max) GSM1800 : 127 mA (700 mA max) GSM1900 : 113 mA (650 mA max)
Puissance d'émission	Class 4 (2 W) pour bandes GSM/EGSM Class 1 (1 W) pour bandes DCS/PCS
Lecteur de carte SIM	Intégré au module, supporte les cartes SIM 3,3v et 1,8v
Antenne	Externe par connecteur MMCX
Interfaces	Connecteur 60 points (CVILUX CBRB060PC2000R0) : Audio, (2x analog, 1x digital), I2C bus, SPI bus, 2x ADC, 2x analog out (PWM), 12 GPIOs et 2 port série de type UART
Normes respectées	GSM07.07 et GSM07.05
Modes SMS	PDU et TEXT
GPRS Data Services	GPRS multi-slot class (MSC) 10 (4+1, 3+2), GPRS PBCCH/PCCCH support, GPRS Class B and CC
FAX	G3, Classe 2.0
Température d'utilisation	– 20 °C à + 55 °C
Taille	33,5 mm x 38,8 mm x 5,6 mm
Masse	< 10 g

Tableau 1 Caractéristiques principales de la carte TM2

Le module TM2 doit être alimenté par une tension de +3,8v via ses entrées VBAT. Compte tenu de l'intensité absorbée par le module GSM notamment lors des phases de recherche de réseau, il est necessaire d'utiliser un bloc alimentation secteur délivrant au moins une intensité de 1A pour une tension continue comprise entre 9 et 12v.

La carte TM2 dispose d'une liaison UART à deux broches RxD, TxD cadencées par défaut à une vitesse de 115 200 bauds pour pouvoir communiquer avec le monde extérieur.

Figure 8 Le module TM2 vu de dessous

Figure 9 Le module TM2 vu de dessus

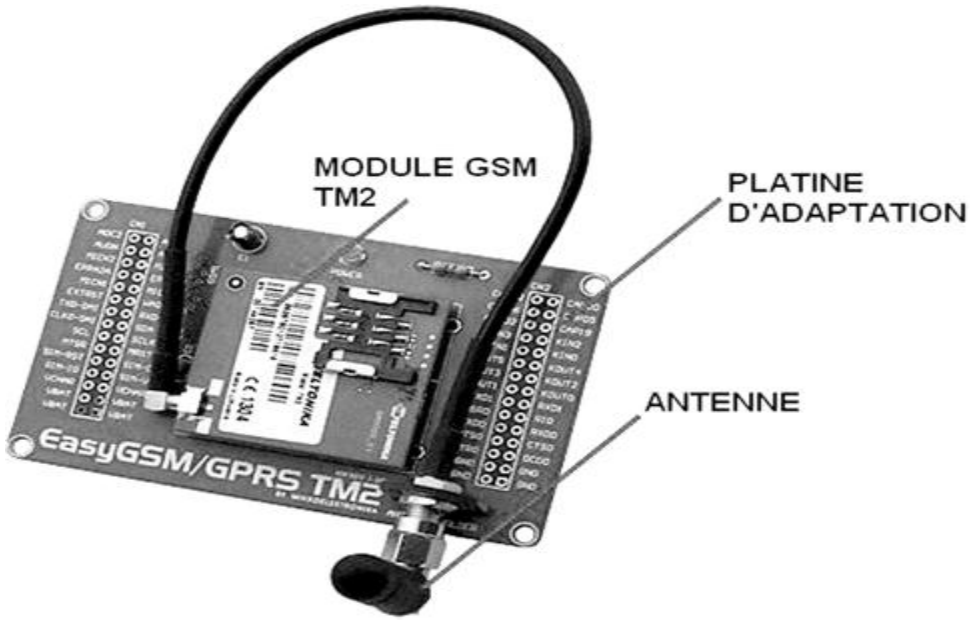


Figure 10 Le module TM2 avec son antenne sur sa platine d'adaptation

Procédé de communication

La communication avec le module Gsm intégré se fait à travers une liste de commandes commençant par les lettres « AT » (on les appellent commandes AT) transmises via un terminal connecté au module. Ces commandes sont sous forme de trois modes comme le montre le tableau suivant :

Commande de test	**AT+CXXX=?**	Retourne la liste des paramètres utilisables avec la commande CXXX.
Commande de lecture	**AT+CXXX?**	Retourne le ou les paramètres en cours associés à la commande CXXX.
Commande d'écriture	**AT+CXXX=<xxx>**	Applique le ou les paramètres <xxx> à la commande CXXX.

Tableau 2 Les trois modes de commandes

III. 4. 3) Présentation de clavier hexadécimale

Le clavier est de type matriciel, quatre lignes et quatre colonnes (4x4), ce qui nécessite huit broches de connexion. Il comporte 16 touches (figure 2) dont 10 pour les chiffres (de 0 à 9) ainsi que les lettres A, B, C, D, E, et F.

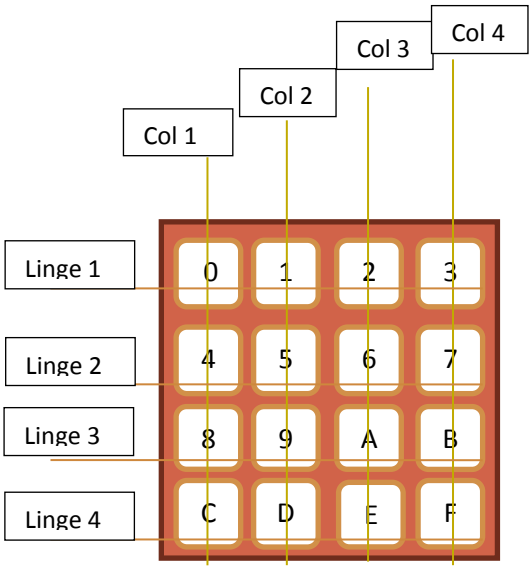


Figure 11 Clavier hexadécimale

Les lignes sont des sorties tandis que les colonnes sont des entrées, l'écoute se fait par scrutation. Le principe est qu'on alimente les lignes une par une a tours de rôle, pour scanner les colonnes, et pouvoir ainsi déceler la touche appuyée.

III. 4. 4) Présentation de l'afficheur LCD [19]

L'afficheur à cristaux liquides est le composant d'affichage le plus utilisé actuellement dans les dispositifs portables. C'est à travers cet afficheur que nous fournissons une interface utilisateur.

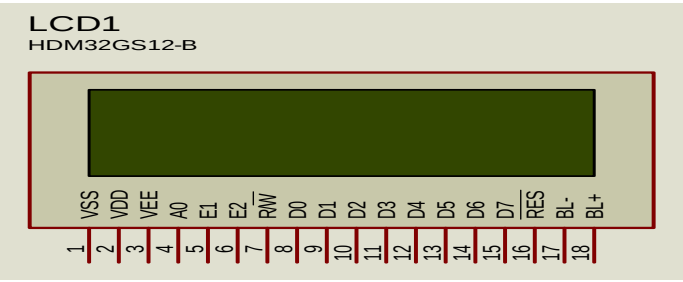


Figure 12 Représentation de l'afficheur LCD

Caractéristiques technique de l'afficheur LCD :

Le dialogue avec un µC se fait par un bus de données de 8 bits ou de 4 bits. Les échanges d'informations sont synchronisés par des signaux de commandes suivant :

- R/*W* (Lecture/écriture) : sélectionner l'instruction lecture ou écriture.
- RS (Register Select) : sélectionner le registre d'instruction (IR), ou alors le registre de donnée(DR).
- E (Enable : Mémorisation) : représente le signale de validation.

Mode de communication

Nous distinguant deux modes de communication illustrée dans la figure 13 :

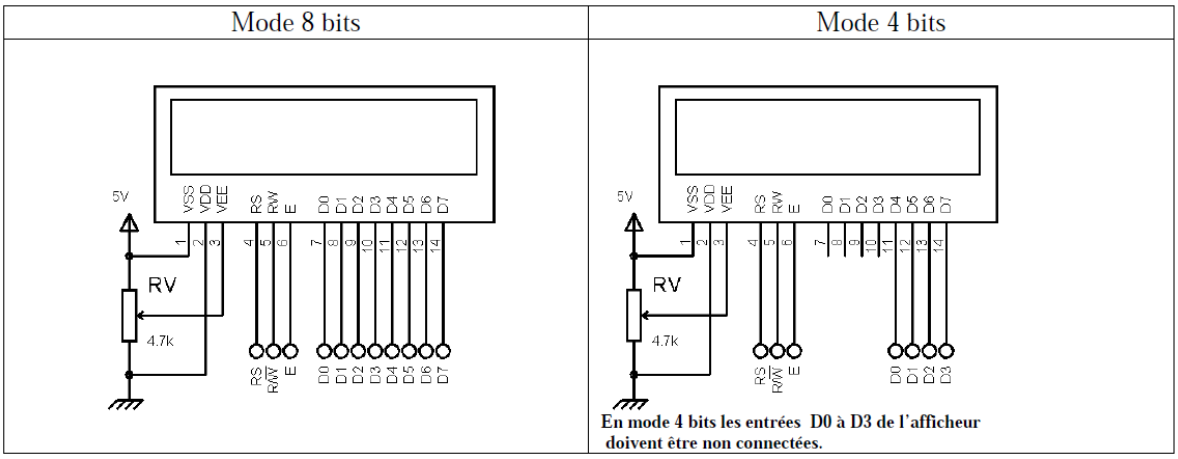


Figure 13 mode de communication avec un afficheur LCD

III. 4. 5) Présentation du détecteur à infrarouge

Figure 14 Détecteur à infrarouge

Le détecteur utilisé est le PIR-PRO 2VI.Ce capteur propose un ensemble de caractéristique :

La portée : 12m*12m.

Alimentation : 8.5 à 15.4Vcc 16mA nominal a 12Vcc, 18mA max.

Relais d'alarme : fermé au repos (sans qu'il y ait détection).

a) Fonctionnement

Le capteur comporte quatre pins. Deux pins spécifiques pour l'alimentation, et deux autres réservés pour l'état de l'alarme.

Alimentation : la Vcc (tension a 12V), et la GND.

Etat de l'alarme :

L'état de l'alarme est donné à travers deux pins constituant un circuit fermé à l'état normal, et ouvert à l'état de détection.

III. 5) Conception Matériel :

Dans cette partie nous allons mettre en œuvre l'interconnexion des différents composants cités plus haut, et l'alimentation qui va servir à faire fonctionner ce système.

Dans un premier temps nous allons présenter la conception matérielle de notre système en donnons l'interface de chaque paire interdépendante des modules. Pour ensuite présenter le schéma global.

III. 5. 1) L'interfaçage des éléments

a) Interface capteur/microcontrôleur :

L'alimentation principale du détecteur à infrarouge va être assurée par un bloc d'alimentation 12V. Le connecteur du contact NF (Normalement fermé) du détecteur est relié à 5V d'une part, et à une broche d'entrées/sortie à travers une résistance de rappel à la masse.

Le contact du détecteur étant normalement fermé, le microcontrôleur reçoit un 1 logique au repos (pas de détection), si un intrus est délecté, le capteur ouvre le contact, envoyant ainsi un niveau logique 0 au port. La figure 15 illustre la connexion entre le microcontrôleur et le détecteur.

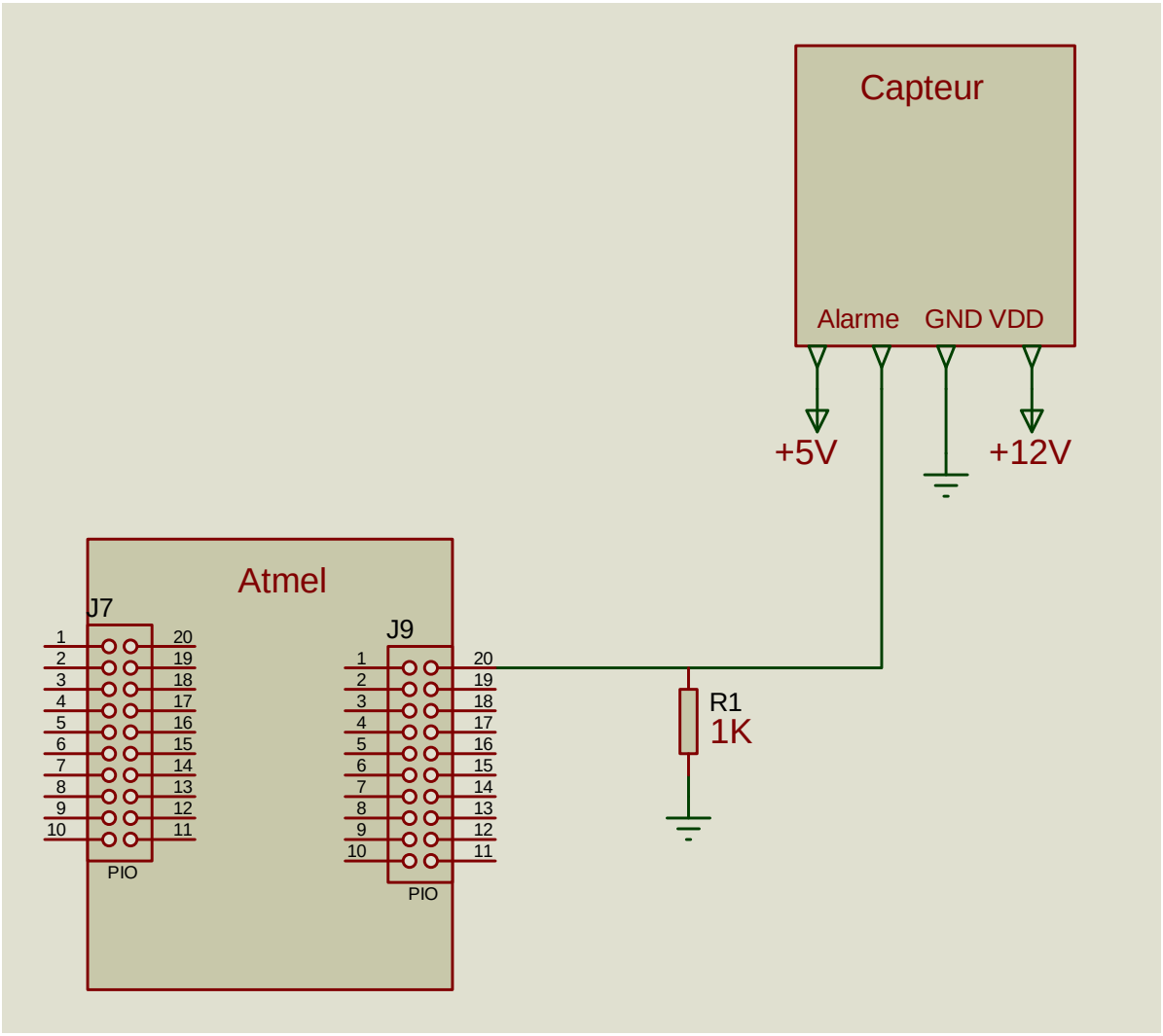


Figure 15 Interface capteur à infrarouge/Microcontrôleur

b) Interface clavier hexadécimal/microcontrôleur :

L'interconnexion entre le clavier matriciel hexadécimal et le microcontrôleur est assurée par un chainage direct entre les huit « 8 » broches du clavier aux ports du microcontrôleur. Le principe de la scrutation est assuré par logiciel qui, grâce à une configuration adéquate des entrées/sorties nous indique la touche enfoncée par l'utilisateur. La figure 16 illustre la connexion entre le clavier et le microcontrôleur.

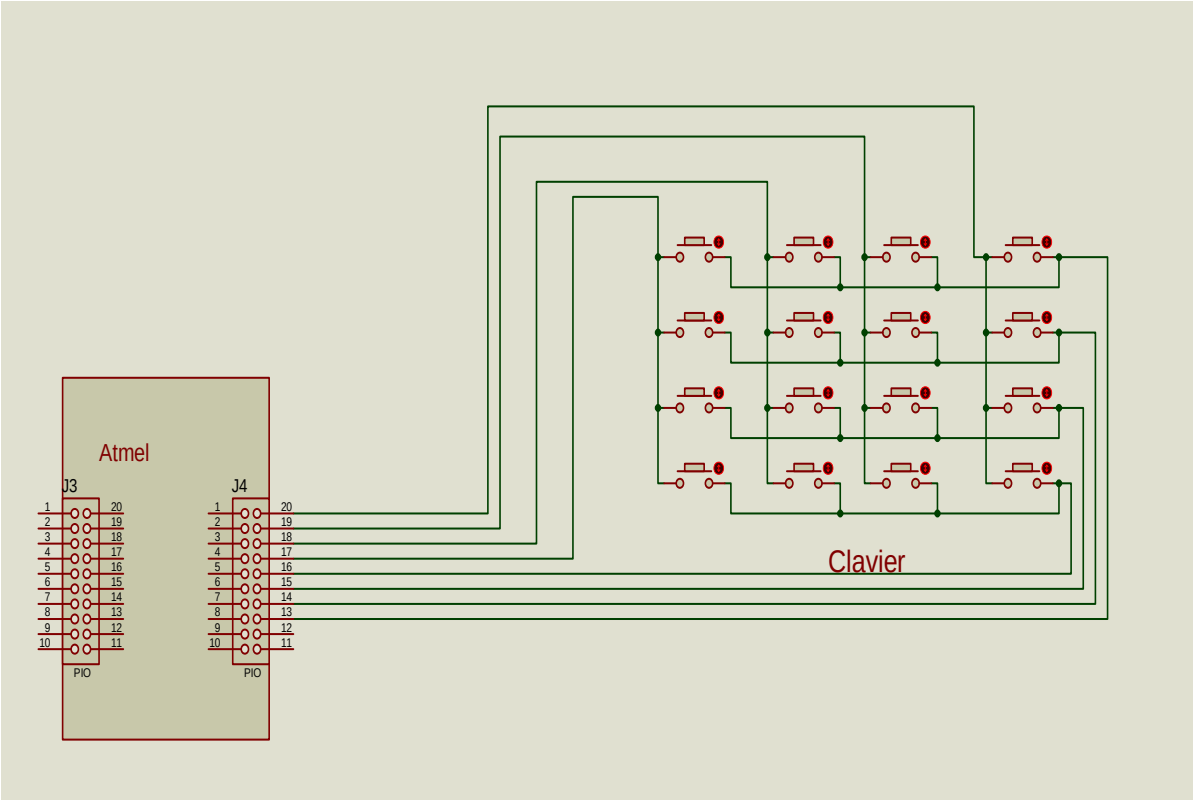


Figure 16 Interface clavier/microcontrôleur

Les touches du clavier sont organisées en 4 lignes et 4 colonnes. Le processeur envoie un niveau logique 1 sur les lignes (une à la fois) et récupère le niveau sur les colonnes pour déduire la touche enfoncée.

c) Interface afficheur LCD/Microcontrôleur

L'implémentation de l'afficheur LCD est conçue comme suit :

L'afficheur LCD (à travers la VDD) est alimenté par une tension de 5V. Les six broches de l'afficheur sont reliées aux ports du microcontrôleur pour le commander en mode de quatre bits (4 bits pour les données plus les lignes RS et E). Une ajustable est ajoutée pour le réglage du contraste de l'afficheur LCD. La figure 17 illustre la connexion entre le microcontrôleur et l'afficheur LCD.

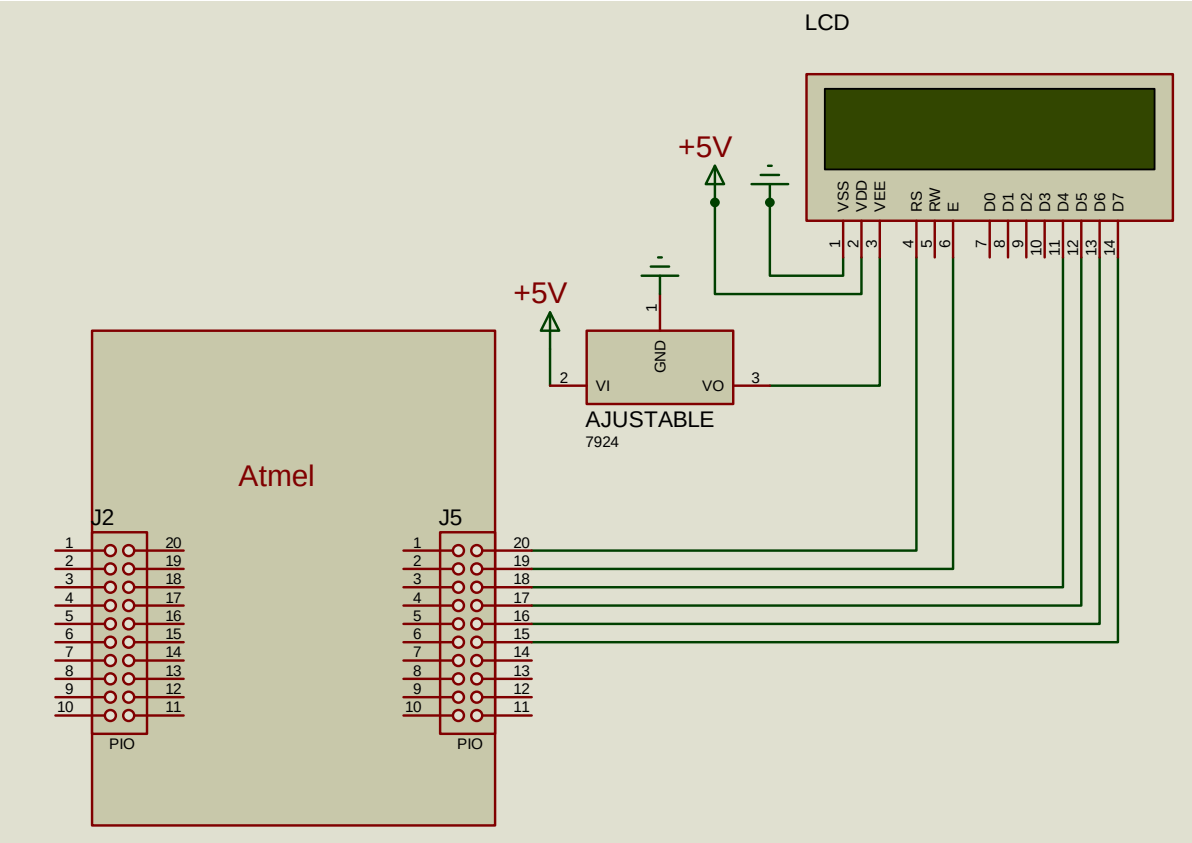


Figure 17 Interface Afficheur LCD/Microcontrôleur

d) Interface Module TM2/Microcontrôleur

Le module TM2 avec, nécessite un interfaçage en mode série UART. Le microcontrôleur dispose de deux interfaces USART. Nous avons utilisé l'une des interfaces (USART0) associées aux broches 21, 22 (TXD, RXD):

Le module TM2 est alimenter à travers la VDD part une tension de 3.8V. Les sorties TXD0, RXD0 du module TM2 sont reliées respectivement aux lignes RX0, TX0 du microcontrôleur

La figure 18 montre l'implémentation du module TM2 avec le microcontrôleur.

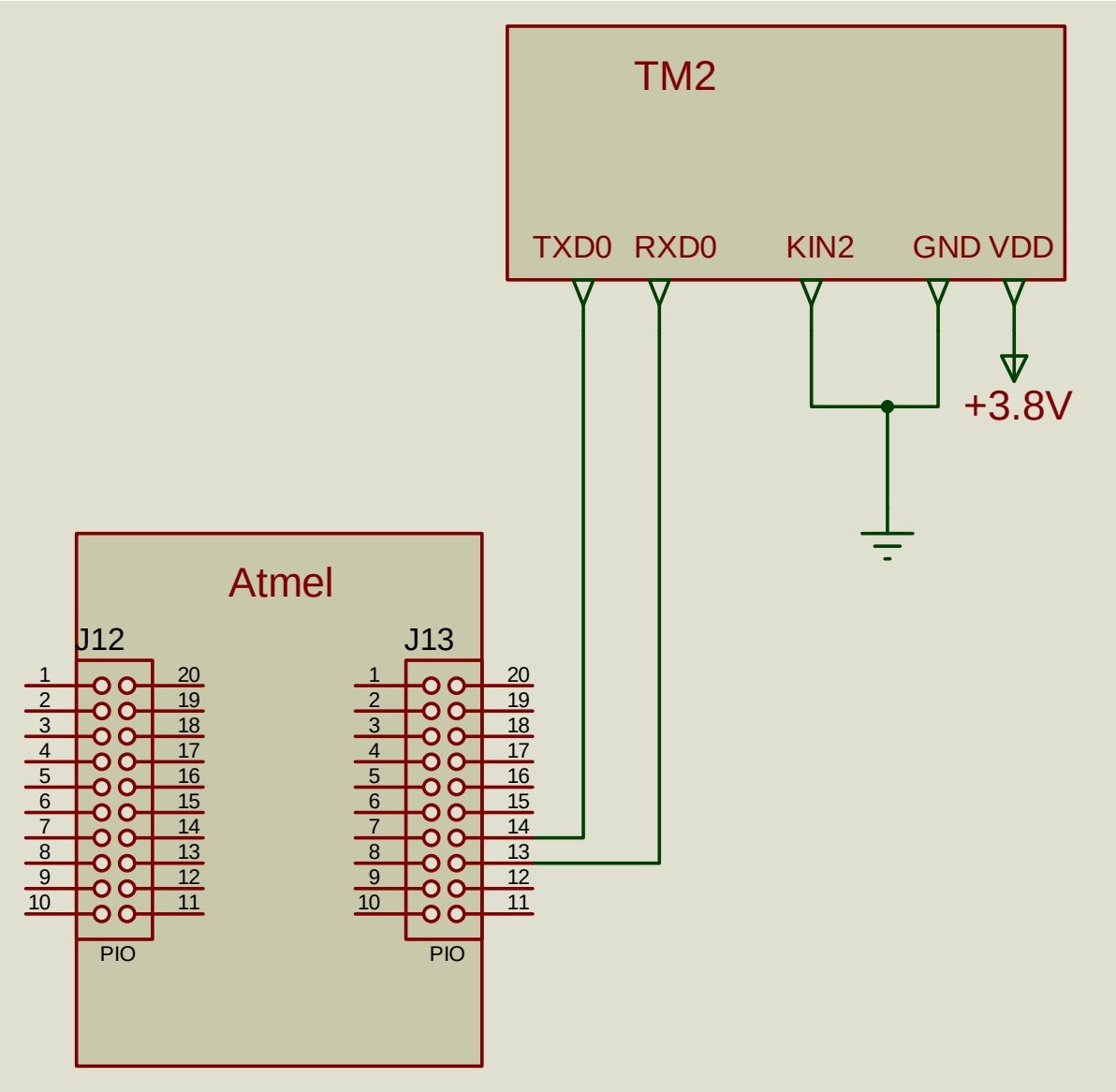


Figure 18 Interface du module TM2 avec le microcontrôleur

Le contrôle du module TM2 est assuré par l'envoie de commandes AT via le port série..

e) Interface de l'alimentation centrale :

Pour le bon fonctionnement de notre système, une alimentation ondulée est nécessaire. Elle va nous fournir les tensions nécessaires, et qui sont :

- Une tension de 5V, pour alimenter le microcontrôleur.
- Une tension de 12V, pour alimenter les capteurs.
- Une tension de 3.8V, pour alimenter le module de transmission.

La composition du bloc d'alimentation est la suivante

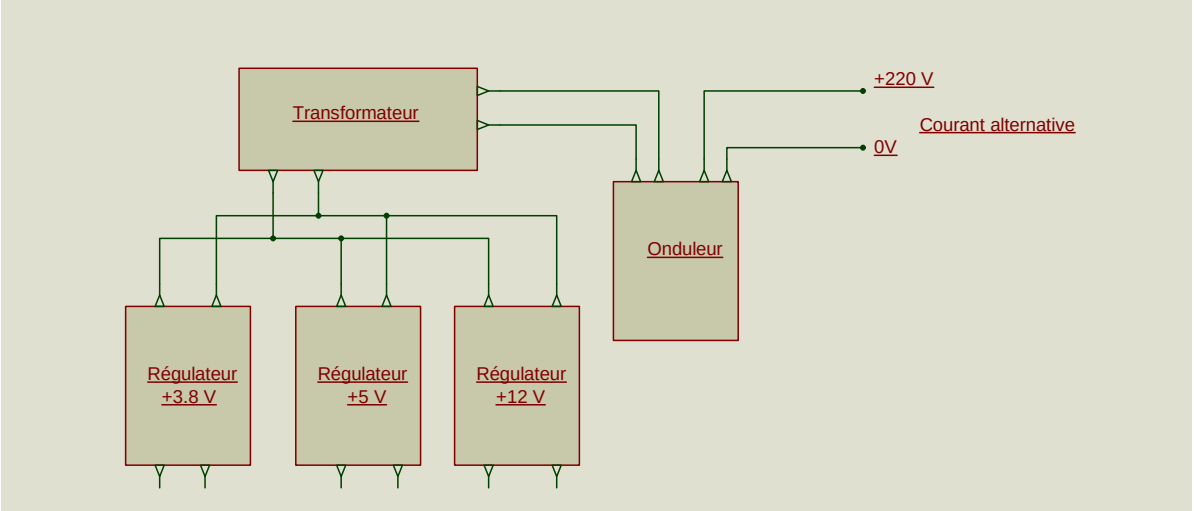


Figure 19 Conception du module d'alimentation pour notre système

f) Interface sirène/microcontrôleur :

L'interfaçage de la sirène et le microcontrôleur se fait comme suit :

On utilise un relais magnétique à deux entrées et une sirène simple. Le circuit d'alimentation de la sirène passe par le relais magnétique. Connecter une des broches du microcontrôleur a à un transistor de commutation relié au circuit d'activation du relais. Quad le port est à 0, le transistor est bloqué coupant ainsi le relais de la sirène. Quand la broche est à 1, le transistor est saturé, permettant ainsi d'activer le relais qui alimente la sirène

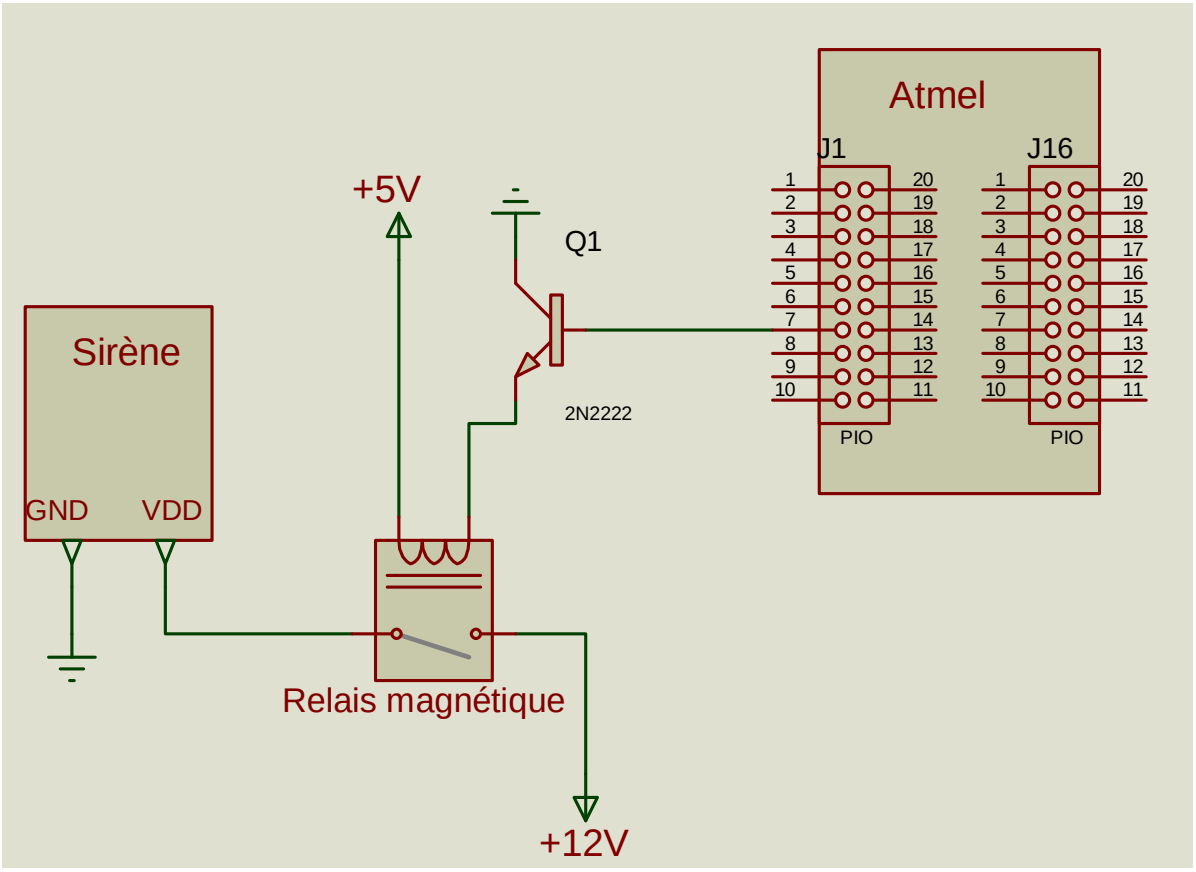


Figure 20 interface microcontrôleur/sirène

Schéma global de notre système

La figure 21 donne le schéma matériel global de notre système embarqué :

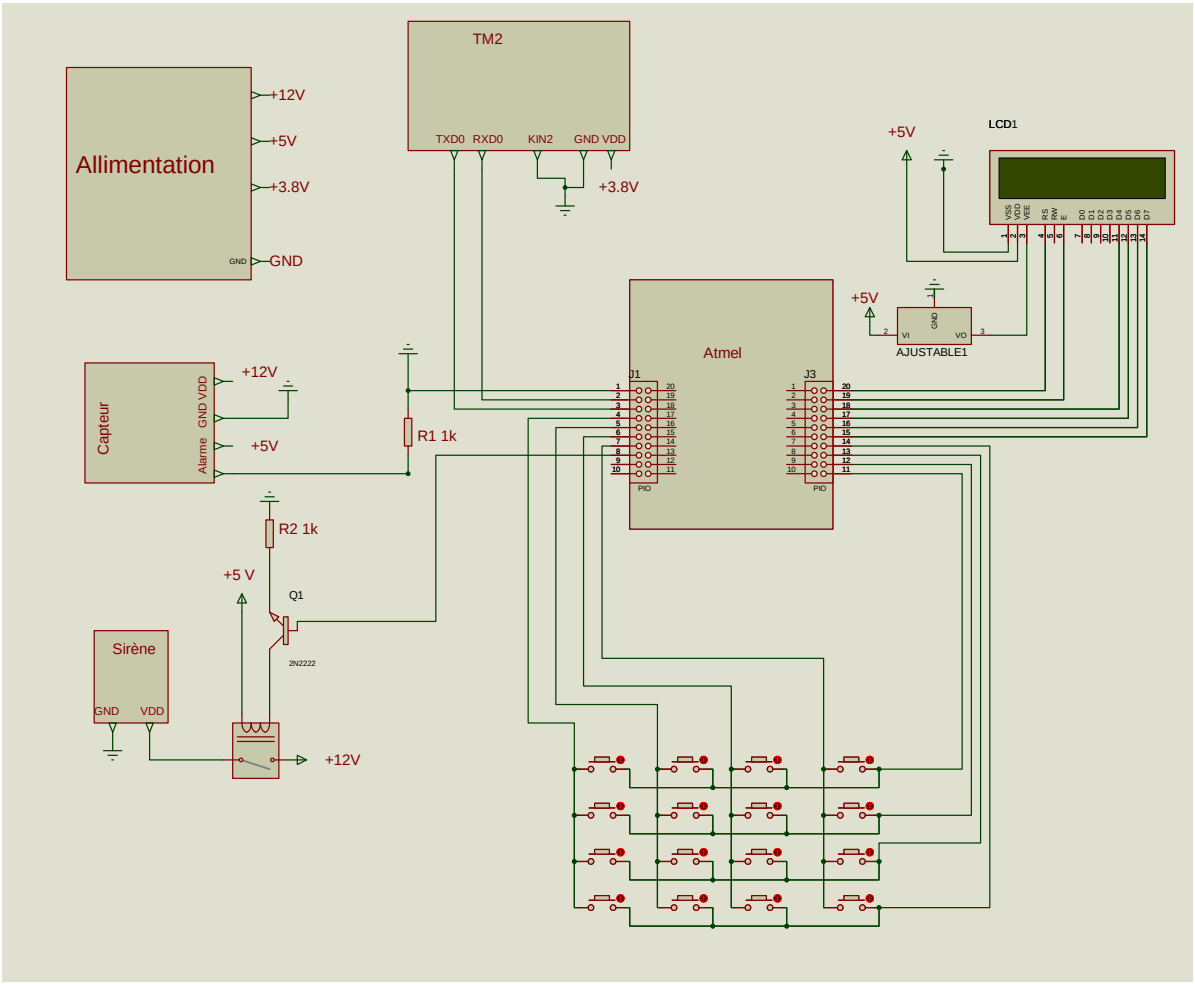


Figure 21 Interface globale de notre système au niveau matérielle

III. 6) Conception logicielle :

La partie logicielle assure le pilotage du matériel. Elle permet entre autres la configuration des broches d'entrées/sortie pour assurer la communication entre les différents modules (Clavier, détecteurs, afficheur,…etc). Elle nous permet particulièrement de configurer le port série communicant avec le module GSM

La procédure de conception de notre logiciel, débute de la prise en considération du matériel disponible, de la définition des tâches à accomplir, la proposition des fonctionnalités utilisateur, jusqu'à la prise en considération du fait que le logiciel représente un large système de contrôle. Notre système représente à lui seul une application finit qui doit satisfaire de lui seul l'ensembles des exigences (sans aucun code ou driver ajouté).

Les taches de notre système embarqué se décomposent en deux catégories. Les taches itératives, et les taches évènementielles (par interruption).

a) Taches itératives:

Les taches à fonctionnement itératif se manifestent par une boucle infinie qui s'exécute tant que le système est sous-alimentation. La scrutation clavier, l'interrogation des capteurs sont des tâches à exécuter de manière itérative

b) Taches évènementielles:

Les taches à fonctionnement évènementiel sont sollicitées par le déclenchement d'un évènement. Dans notre cas le calcul de la base de temps (1 ms) est une tâche évènementielle causée par l'interruption du timer qui nous sert à définir toutes les temporisations nécessaires au fonctionnement de notre système.

III. 6 . 1) Fonctionnement global de notre logiciel

Dans notre logiciel nous distinguons quatre états différents « Désactivée » « surveillance » « Temporisation » « Alerte »

Nous proposons dans cette figure 22 le diagramme de transition des d'états possibles

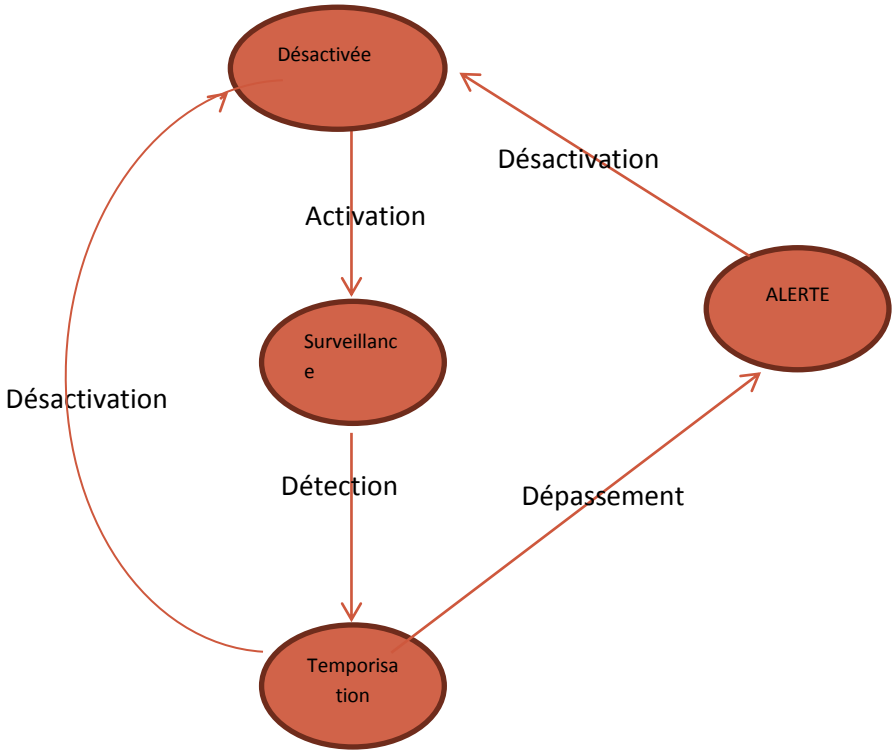


Figure 22 diagramme de transition des d'états possibles

Chaque état comporte un ensemble de taches. Certaines sont propres à l'état d'autres sont communes à plusieurs états.

III. 6 . 2) Description des états

a) Etat désactivé

Dans cet état l'alarme ne surveille pas les capteurs. Elle boucle en exécutaun une seule tâche qui est la scrutation clavier en attente d'un code d'authentification qui fera passer l'alarme dans l'état surveillance

```
Alarme_desact(){
    Tantque (état=desactivé) faire
        Sucruter_clavier(mot_de_passe)
        Si (mot_de_passe_correct) alors
          Etat ←activée
       finsi
  }
```

b) Etat surveillance

Dans cet état, l'alarme continue à scruter le clavier en attente du mot de passe de désactivation, et scrute également les capteurs en attente d'une éventuelle intrusion.

Si un mot de passe correct est introduit, l'alarme est désactivée. Si un intrus est détecté, l'alarme passe dans le mode Temporisation

```
Alarme_surveil(){
    Tantque (état=surveil) faire
        Sucruter_clavier(mot_de_passe)
        Si (mot_de_passe_correct) alors
          Etat ←désactivé
       finsi
        scruter_capteurs()
        Si (intrusion_détectée) alors
          Etat ←Temporisation
        Finsi
  }
```

c) Etat temporisation

Dans cet état l'alarme est en attente du code de désactivation (en cas ou la personne détectée est le propriétaire). Si le code n'est pas introduit au bout de la temporisation prédéfinie, l'alarme passe à l'état Alerte.

```
Alarme_Tempo(){
   Tantque (état=Tempo) faire
        Sucruter_clavier(mot_de_passe)
        Si (mot_de_passe_correct) alors
          Etat ←désactivé
       finsi
        control_Tempo()
        Si (Tempo_dépassé) alors
          Etat ←Alerte
        Finsi
  }
```

d) état Alerte

Dans cet état, l'alarme se déclenche. Elle active la sirène et elle envoie un SMS via le TM2. Elle reste dans cet état jusqu'à introduction du mot de passe

```
 Alarme_Alerte(){

        Déclencher_sirène()
        Envoie_SMS()
      Tantque (état=Alerte) faire
        Scruter_clavier()
        Si (mot_de_passe_correct) alors
          Etat ←désactivé
       finsi
        FinTanque
Finsi
  }
```

III. 6 . 3) Description des tâches

a) Scrutation du clavier

Les quatre lignes du clavier hexadécimal sont considérées comme sorties du microcontrôleur, et les colonnes comme entrées pour être écoutés par ce dernière. La structure matricielle du clavier permet à travers un appuie sur une touche, d'interconnecter la ligne et la colonne permettant de véhiculer l'état du signal de la sortie vers l'entrés. De ce fait le principe de la scrutation du clavier consiste à alimenter les lignes de sorties une par une et à tours de rôle, et rester à l'écoute des colonnes.

Algorithme de scrutation de clavier

```
L1, L2, L3,  L4 les quatre lignes  qui sont en sorties

C1, C2, C3, C4 les quatre colonnes  en entrées

Scrutation(touche){
Début
   N= 3
   M= 3
   Table [N,M]=[[0,1,2,3], [4,5,6,7], [8,9,A,B], [C,D,E,F]]
    L0←0, L1<=0, L2<=0, L3<=0
         Pour (i<=0 à 3)
         Li<=1
                  Pour (j<=0 à 3)
                  Début
                           Si  (Cj =1)
                           debut
                              touche <=table[i,j]
                           fin

                  fin
fin
}
```

b) Scrutation de capteur

Le fonctionnement de la tâche de la scrutation des capteurs est assez simple. L'idée consiste à une écoute des capteurs un par un. (Une écoute parallèle biens qu'envisageable offre des contrainte notamment pour déceler la quel des entrées a été perturbé)

Algorithme de scrutation des capteurs

B1, B2, B3, B4 les quatre broches de capteurs et qui sont des entrées

Scrutation_capteur©

Début

Pour (i<=0 à 3)

Faire

Si (Ci =1) alors

C ← ci

finsi

fait

fin

c) Affichage sur LCD

Pour faire fonctionner l'afficheur LCD, nous avons programmé des routines afin de pouvoir le piloter, entre autre :

- Effacer l'écran.
- Positionnement du curseur.
- Mode d'affichage (curseur visible, clignotement du curseur).
- Mode de fonctionnement (mode quatre bits, mode huit bits).
- Affichage

Initialisation de l'afficheur en mode quatre bits :

Afin d'initialiser l'afficheur en mode quatre bits, un ensemble d'instructions sont exécutées selon l'organigramme suivant.

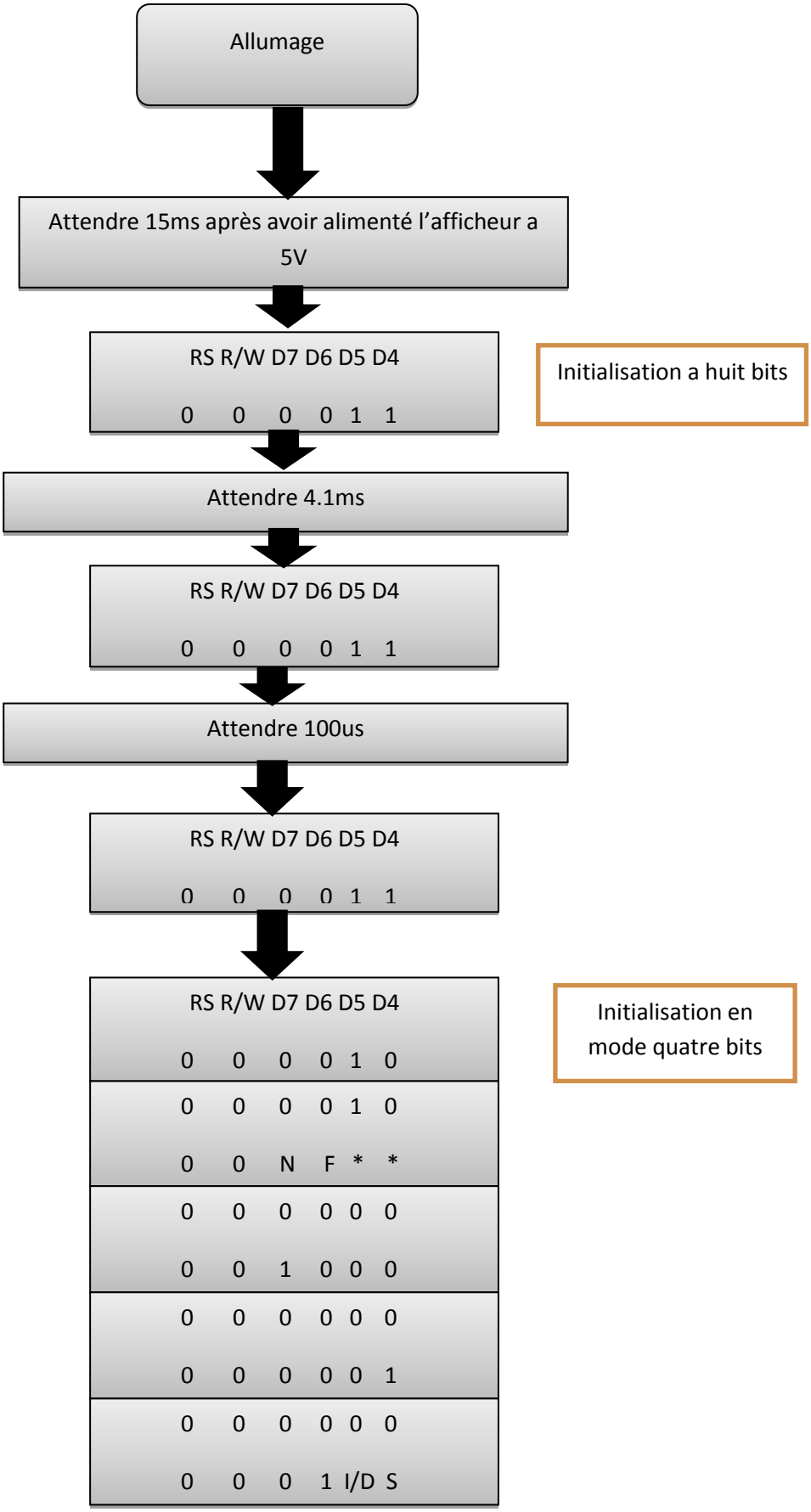


Figure 23 Diagramme d'initialisation en mode quatre bits

d) Routine du timer :

La routine du timer fonctionne en mode interruption. Nous avons programmé ce dernier à générer une interruption chaque 1 ms. La routine du timer consiste à incrémenter la variable milli et décrémenter la variable de temporisation éventuelle appropriée

Temporisation

La temporisation repose sur la routine du timer qui calcule la base de temps. Une variable tempo est intitialiséée à la valeur souhaité. Chaque interruption du timer, cette dernière est décrémenté. Une fois arrivée à 0 la temporisation est terminée.

Algorithme de temporisation :

Contrôle_tempo()

```
Debut
Tempo ← tempo_souhaitée
    Tant que tempo > 0 faire
        Debut
        Aller a scrutation du clavier
        Si mot de passe = correcte alors
         Aller à alarme_désactivée()
        Fin si
    Fin tant que
    Tempo_sépassée ←1 ;
fin
```

III. 6 . 4) Résumé du fonctionnement du système

Dans cette partie nous allons résumer par un organigramme les différentes fonctions de notre système

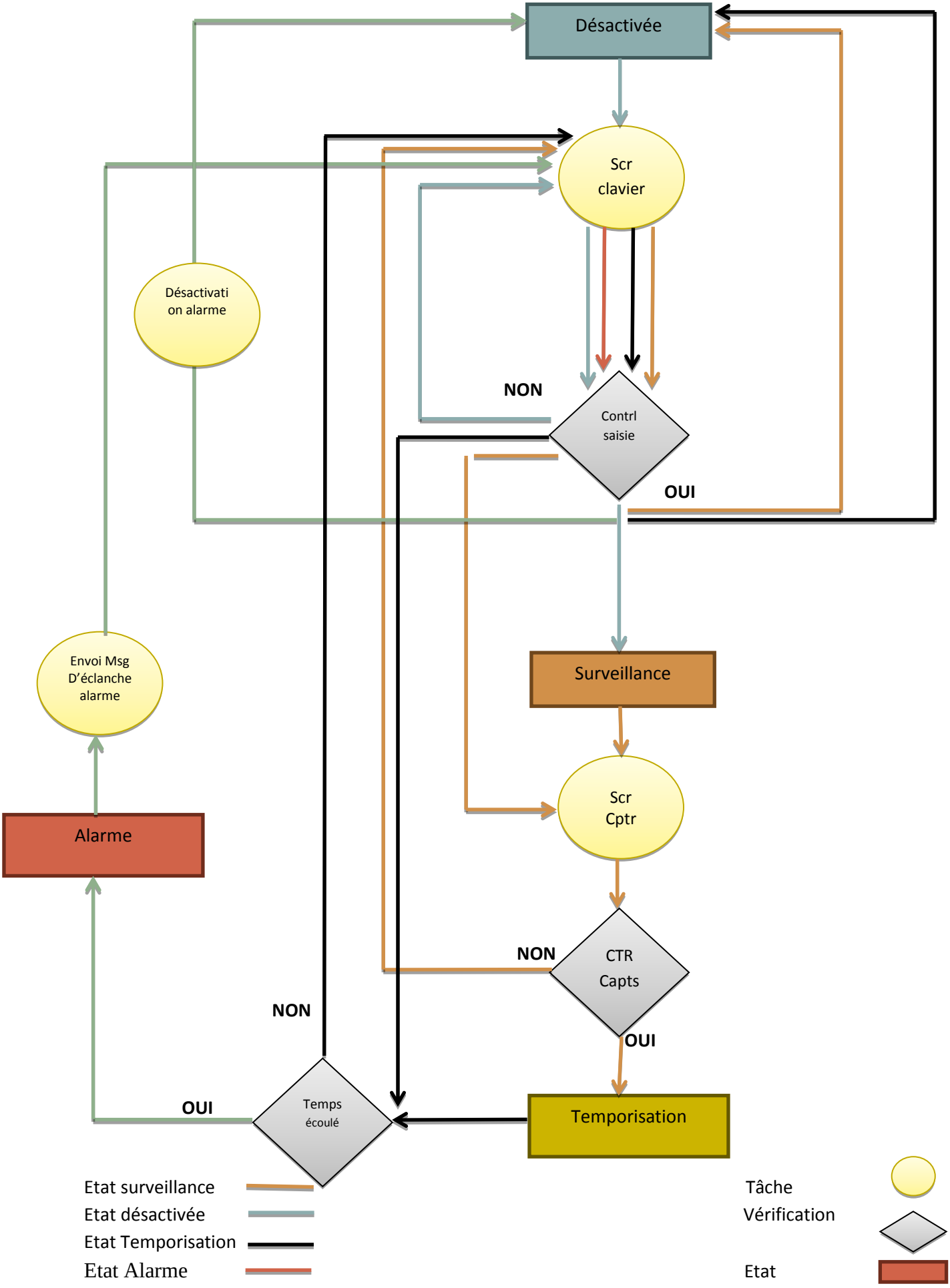


Figure 24 Diagramme représentant fonctionnement interne

Pour passer de l'état désactivé à l'état surveillance, le système vérifie continuellement le clavier. Une éventuelle saisie sur ce dernier éveille la tache vérification mot de passe pour soit passer à l'état surveillance ou alors rester dans l'état courante tout en affichant un message d'erreur. Notant que trois faussent tentatives mettent le système en blocage temporaire d'une période qui est égale à deux minutes.

Une fois que l'alarme est activée (état surveillance), le fonctionnement est comme suite :

Le système vérifie alternativement les capteurs ainsi que le clavier à travers les tâches assignées à ces deux derniers. La saisie du mot de passe engendre une vérification de l'authentification par le système. Une fois l'authentification y ait, le système passe à l'état désactivé. Sinon un message d'erreur est affiché et le système continue son fonctionnement. Une éventuelle perturbation au niveau des capteurs active l'état d'alerte.

A l'état temporisation le système active la temporisation nécessaire pour une éventuelle authentification. Cette dernière étant dans le lapse de temps prédétermine pour alors nous mener ou bien à une désactivation de système ou alors un simple affichage d'un message d'erreur. Une fois la temporisation arrive a sont échelant le système passe à l'état Alarme

A l'état Alarme le système enclenche la sirène et envoie un message, et vérifie continuellement le clavier.

Notant que trois faussent tentatives déclenchent directement l'envoie du message et l'activation de la sirène.

III. 7) Conclusion

Dans ce chapitre, nous avons passé en revue l'architecture matérielle, et logicielle de notre système de surveillance et contrôle d'accès ainsi que les principales caractéristiques des composants qui vont le constituer et pour lesquels, nous avons justifié leurs sélections. De même, nous avons présenté leurs configurations externes et internes et la manière avec laquelle, chacun d'eux, peut être connecté avec les autres.

Chapitre 4

Réalisation matériel et logiciel

IV. 1) Introduction

Apres avoir cerné nos besoins dans le chapitre précédent, nous allons mettre en pratique ce dernier point.

IV. 2) Outils utilisés

La mise en pratique de notre conception nous contraint à faire appel à un ensemble d'outils physique et logiciels.

a) Keil

Le simulateur keil à travers les diverses bibliothèques dont il dispose, donne un sens la notion du co-design appelé aussi co-développement. Grace à ce dernier nous pouvant concevoir notre application d'une manière indépendante par rapport à l'avancement de la conception hardware.

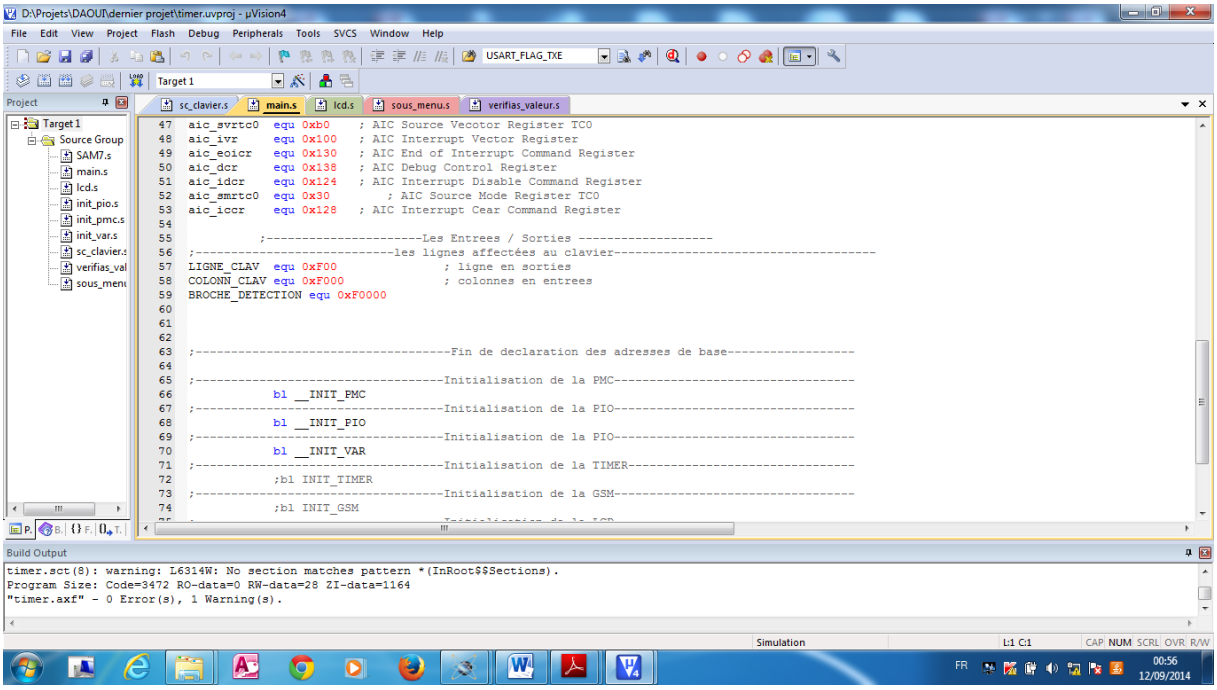


Figure 25 Interface simulateur Kiel

b) Proteus

L'outil Proteus est référencé fréquemment par les électroniciens. Ses possibilités démarrent de la conception des schémas électroniques, de la mise en œuvre des circuits imprimé jusqu'à la simulation de système.

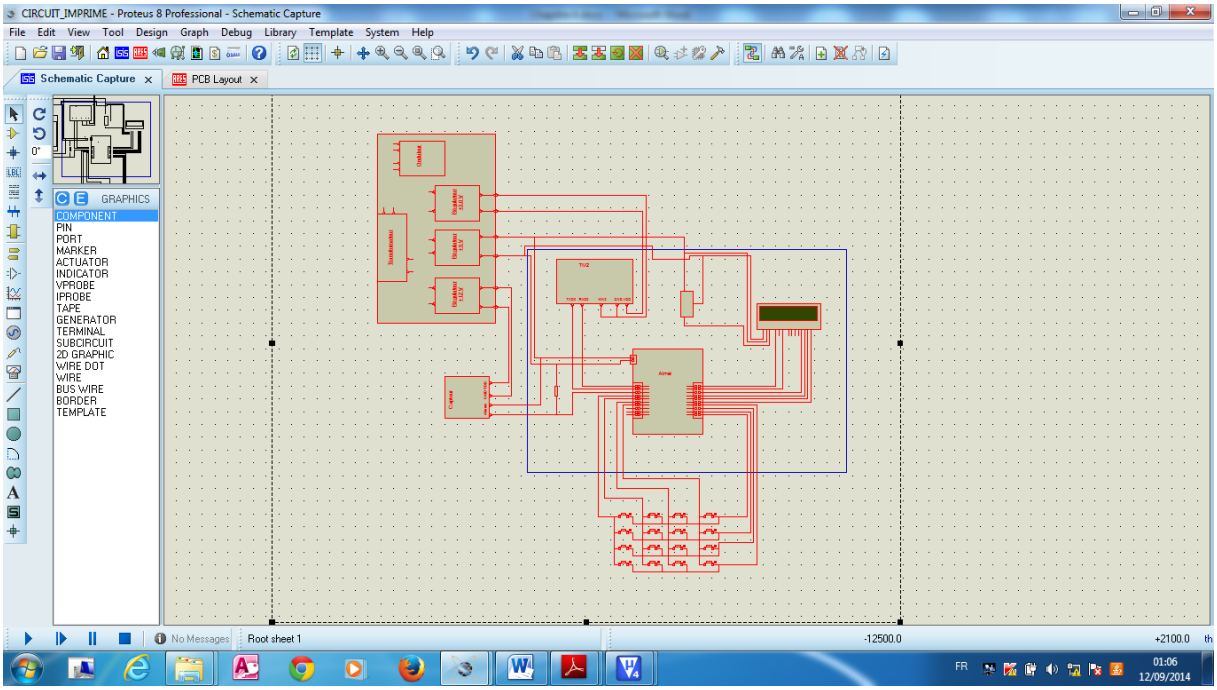


Figure 26 interface du simulateur Proteus

IV. 3) Déroulement de la réalisation

Le processus de réalisation s'effectue en couple matériel et logiciel. Pour chaque interface microcontrôleur module périphérique, une fois câblé, Une partie logiciel propre à cette interface est implémentée et testée pour arriver à une réalisation finale du projet. Pas à pas le montage est d'abord construit sur une carte de montage expérimental pour arriver à un assemblage finale.

Brochage du Microcontrôleur/Périphériques

Nous allons consacrer cette partie aux divers brochages entre le microcontrôleur et ses périphériques. Notons qu'une sélection restrictive et/ou sélective peut être imposée par l'utilisation des ressources internes de notre microcontrôleur. Ce dernier point fait référence au multiplexage de ces ressources avec les broches d'entrées/sorties de la PIO (ex : PIO21 et PIO22, multiplexé avec l'UART1).

La figure 27 donne l'implantation des broches du microcontrôleur.

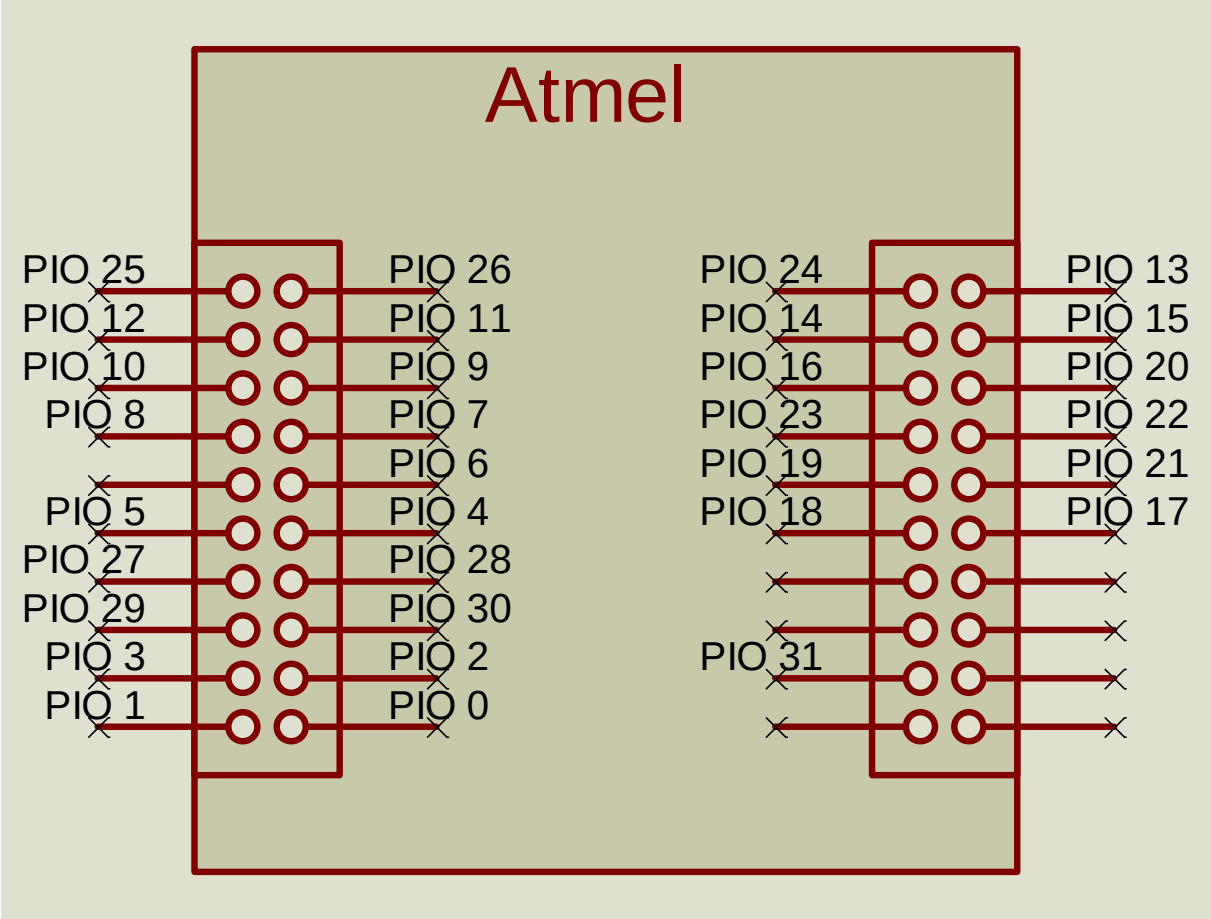


Figure 27 positionnements des broches du microcontrôleur

Le tableau 3 fait correspondre le numéro de la PIO avec les broches des modules périphériques.

PIO Microcontrôleur	Broches périphériques
	Afficheur LCD
PIO_0	D4
PIO_1	D5
PIO_2	D6
PIO_3	D7
PIO_4	E
PIO_5	RS
	Clavier
PIO_8	LIGNE_1
PIO_9	LIGNE_2
PIO_10	LIGNE_3
PIO_11	LIGNE_4
PIO_12	COLONNE_1

PIO_13	COLONNE_2
PIO_14	COLONNE_3
PIO_15	COLONNE_4
	TM2
PIO_21 : RXD1	TXD0
PIO_22 : TXD1	RXD0
	Capteurs
PIO_16	C1
PIO_17	C2
PIO_18	C3
PIO_19	C4
PIO_31	Sirène

Tableau 3 Correspondance des numéros de la PIO avec les broches des modules

La liaison de l'ensemble des périphériques clavier, module réseau, capteurs à infrarouge et afficheur LCD avec un microcontrôleur nous, a contraints à concevoir une carte circuit imprimé pour réduire le câblage. Réalisée grâce à un outil Proteus,

Procédé de conception de la carte PCB

Le procédé de conception démarre de l'outil Proteus, qui en introduisons les informations sur l'emplacement des composants et le chainage entre leurs entrées/sorties, nous propose un choix sur la réalisation d'une carte a une seul couche ou à deux couches. Si dans un premier temps notre choix c'est porté sur une implémentation a une seul couche, le nombre des entrées/sorties qui combiné avec leurs emplacements prédéfinie par nous-même, nous contrains à revoir notre choix du fait de la présence d'un nombre d'interconnexion non désirés entre des pistes.

La figure 28 proposée par l'outil Proteus illustre le pistage entre les lignes des entrées/sorties des divers périphériques.

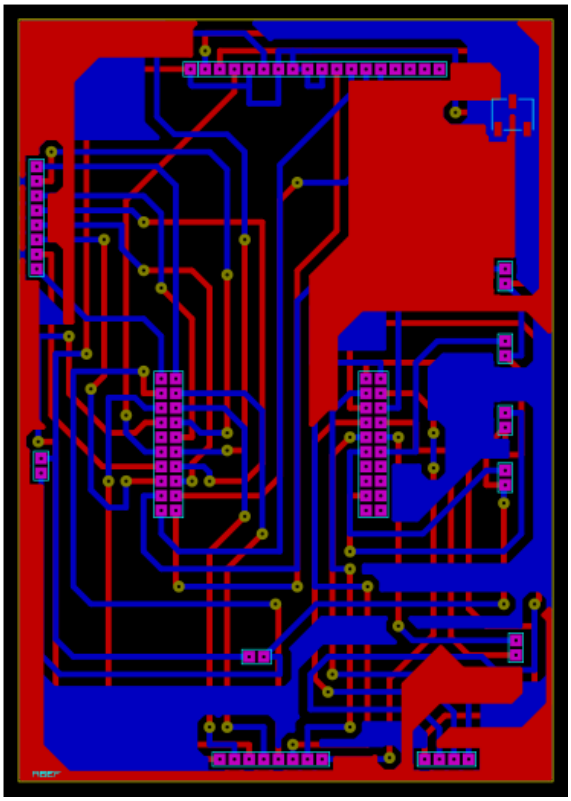


Figure 28 le pistage entre les lignes des entrées/sorties des divers périphériques

La figure 29 représente la réalisation de la carte PCB.

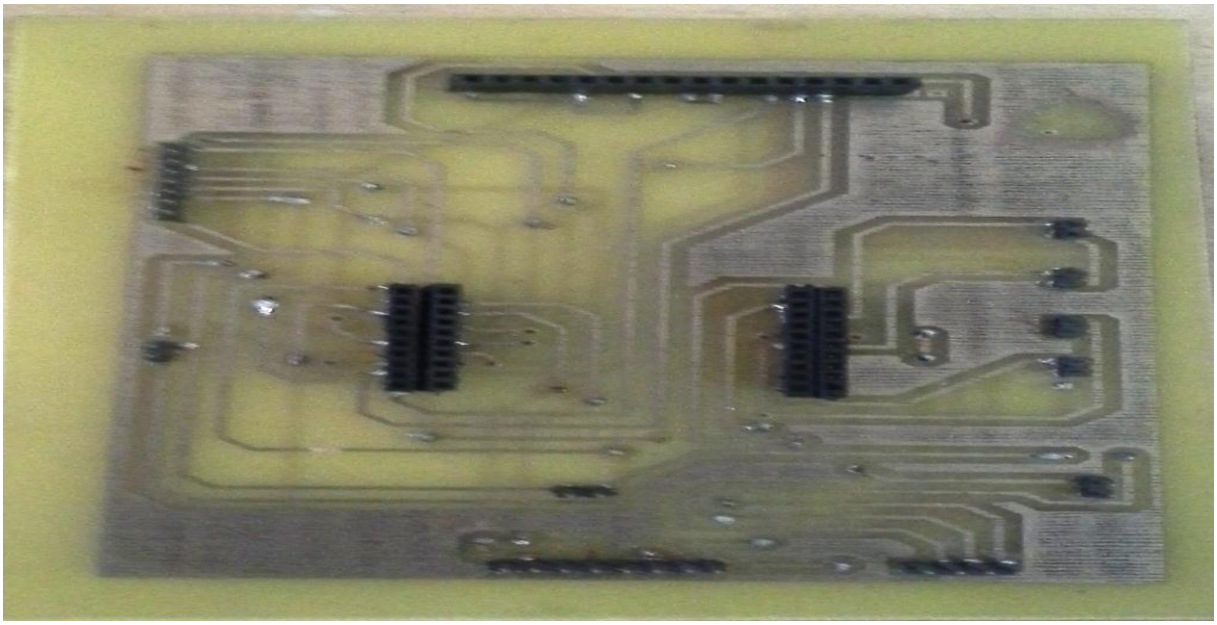


Figure 29 Carte circuit imprimé (PCB)

IV. 3. 1) Réalisation logiciel

La partie logicielle consiste en un programme développé à l'aide du logiciel Keil, et qui sera exécuté par le microcontrôleur. Le chargement de ce programme sur le microcontrôleur est effectué grâce un JTAG.

a) Le programme permet à l'utilisateur de

- Introduire le mot de passe pour avoir accès au lieu protégé.
- Changer le mot passe par les personnes possédant déjà le mot de passe.
- Afficher sur l'écran les différents messages d'invitation et/ou de réponse.
- Activer ou désactiver le système de surveillance.
- Déclencher l'alarme après trois essais erronés de saisie de code.
- Changement de temps pour temporisation.

b) Déroulement de processus

1. Initialisation de la PIO
2. Initialisation de l'afficheur LCD
3. Initialisation des variables
4. Initialisation du PMC (alimentation des périphériques interne du microcontrôleur).
5. Initialisation de l'état de l'alarme

Après les initialisations citées ci-haut, l'afficheur LCD affiche l'état de l'alarme (désactiver). La figure 30 illustre l'état du système à travers l'afficheur LCD.

Figure 30 Affichage d'état Désactivé

L'utilisateur à travers son mot de passe peut activer le système. La saisie du mot de passe donne deux cas de figure :

cas1 : mot de passe erroné, ce qui fait, un affichage du message « erreur » pendant deux secondes comme mentionné dans la figure 31.

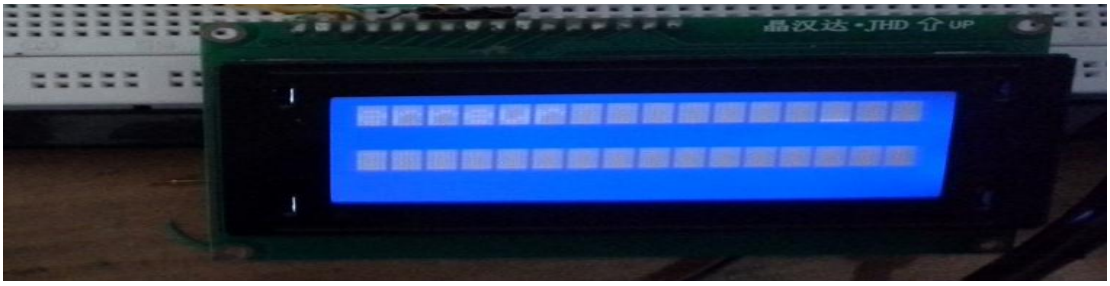


Figure 31 Affichage d'un message Erreur

Une fois les deux secondes sont comptées, un affichage de l'état courant de l'alarme « Désactivée » ce fait, par l'afficheur LCD.

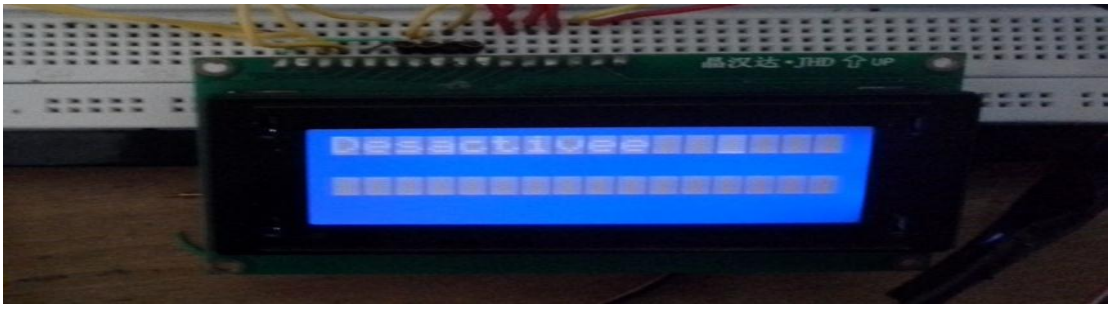


Figure 32 Affichage d'état Désactivé

cas2 : mot de passe validé, donc le système passe à l'état « Surveillance », tout en affichant comme illustré dans la figure 33 l'état courant du système.

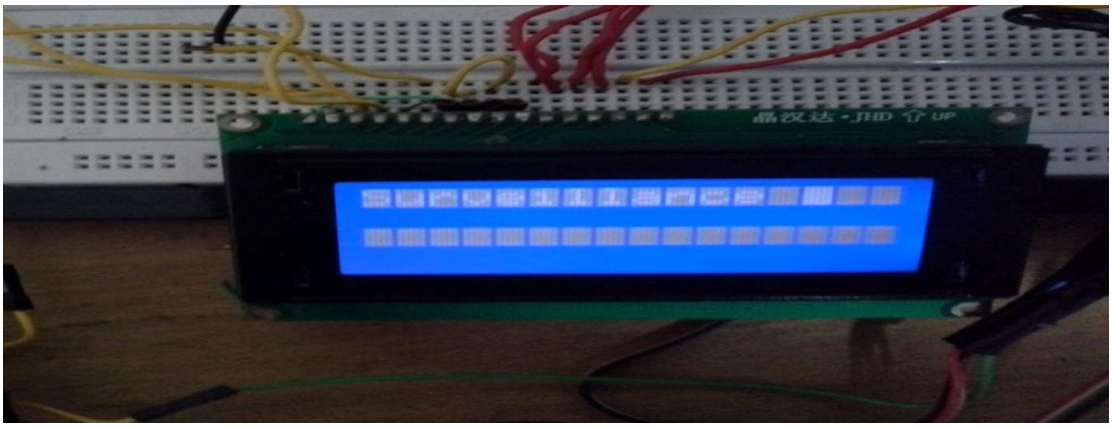


Figure 33 Affichage d'état Surveillance

Dans le cas où le système est en surveillance, une perturbation au niveau des capteurs est détectée ; l'état du système passe à l'état temporisation, tout en affichant le message « Temporisation », dont figure 34 fait référence.

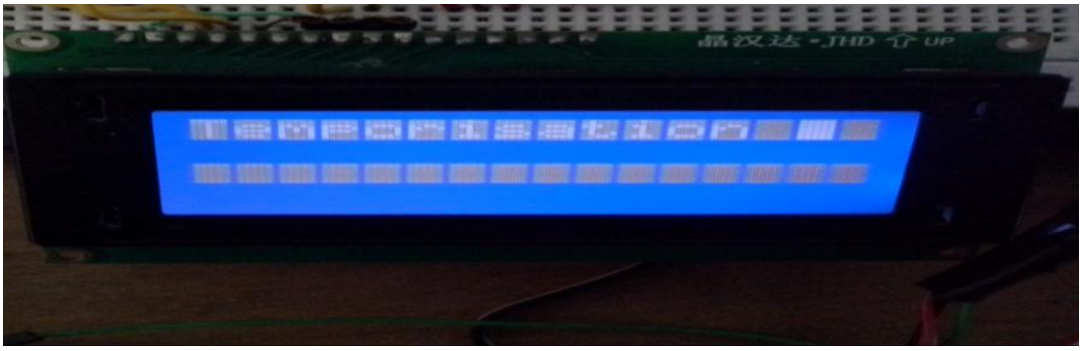


Figure 34 Affichage d'état Temporisation

L'utilisateur peut désactiver l'alarme à travers une authentification par son mot de passe.

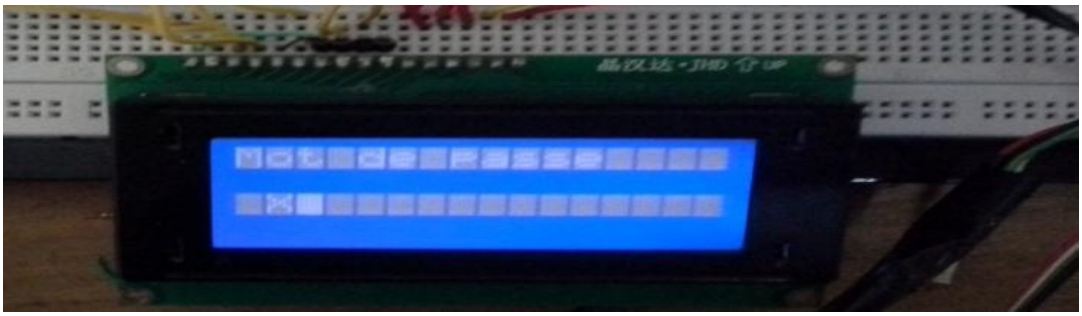


Figure 35 Affichage de la saisie du mot de passe

IV. 3. 2) Fonctionnalités proposés par notre système

Changement du mot de passe

L'utilisateur peut à sa guise changer son mot de passe et cela grâce à la touche préprogrammée « D » dont le fonctionnement est :

Affichage du message « ancien M_D_P », comme le montre la figure 36.

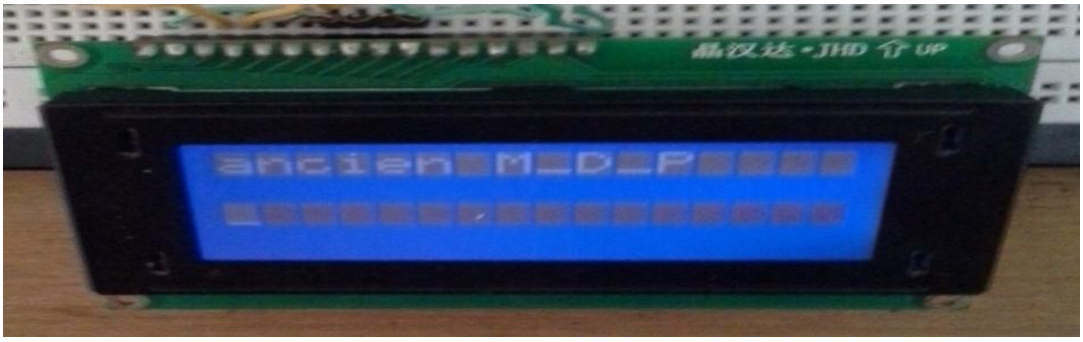


Figure 36 Affichage Ancien mot de passe

Une fois le mot de passe est saisi, le système après control propose deux cas de figures. Cas 1 : mot de passe erroné, et l'erreur est mentionnée à travers l'afficheur LCD, comme l'illustre la figure 37.

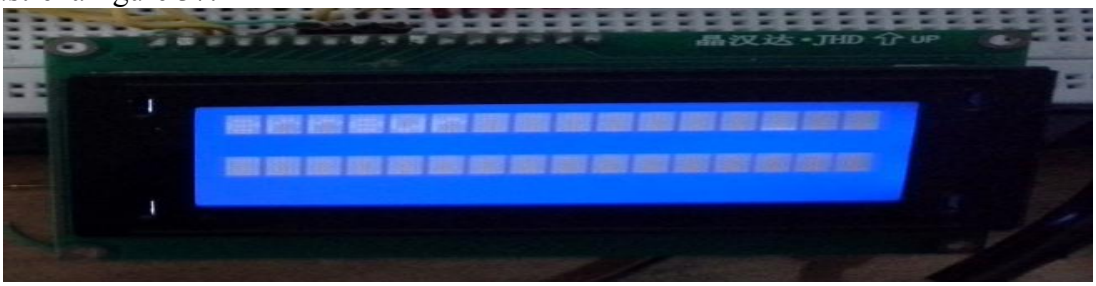


Figure 37 Affichage d'un message erreur

Cas 2 : mot de passe correcte. Le système propose l'insertion du nouveau mot de passe. La figure 38 fait référence à ce cas de figure.

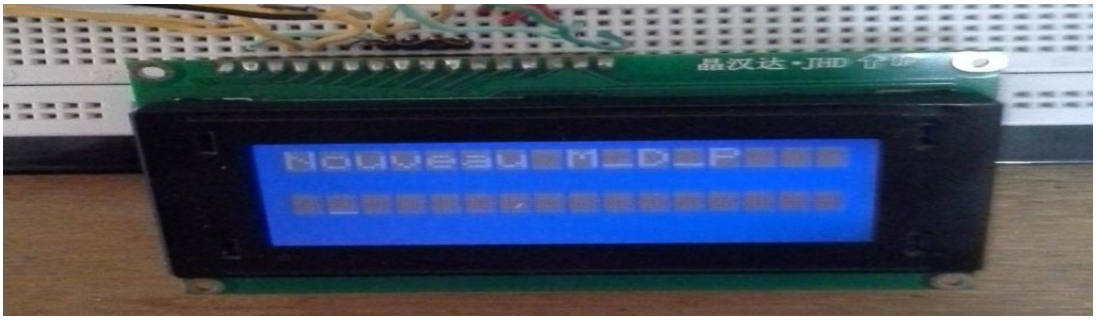


Figure 38 Affichage Nouveau mot de passe

Une fois la saisie est faite, une confirmation du nouveau mot de passe est nécessaire.

Figure 39 Affichage de la confirmation

La conformité donne deux cas de figures :

Cas1 : non-conformité des deux mots de passe ; le système mentionne cet écueil par l'afficheur LCD.

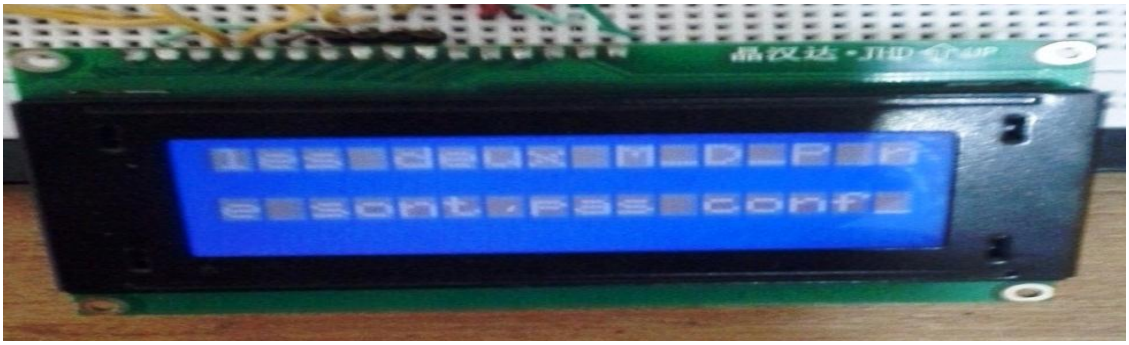


Figure 40 Affichage de la non-conformité

Cas2 : conformité des deux mots de passe, ce qui engendre l'affichage du message « mot de passe inséré », comme illustré par la figure 41.

Figure 41 Affichage l'insertion de mot de passe

Changement du temps de temporisation

La fonctionnalité changement de temps de temporisation ne répond au besoin, que si le système est en mode désactivée. Accessible via la touche « C » du clavier. Cette dernière une fois sollicitée, le système demande le mot de passe pour exécuter la fonctionnalité.

Elle propose trois temporisations (une minute, deux minutes ou trois minutes) configurable à travers respectivement les touches 1, 2, 3 du clavier, comme illustré dans la figure 42.

Figure 42 Affichage les trois temps d'attente

Une fois le choix est mentionné au système, le message « temps attente changé » est affiché sur l'écran LCD.

IV. 4) Conclusion

Dans ce chapitre, nous avons décrit le processus de la réalisation matériel et logiciel du prototype de notre application. Attente

Conclusion

❖ Conclusion général

Dès l'aube de l'humanité, la recherche systématique de la protection a toujours été une fin en soi.

Dans ce projet nous nous somme donné la tâche de concevoir et réaliser un système de surveillance des biens privé.

Le travail présenté dans ce mémoire nous a permis de s'affranchir du procédé de conception des systèmes embarqués, de la partie matérielle à la partie logiciel. Comprenant le fonctionnement d'un microcontrôleur, d'un afficheur LCD, des capteurs et la carte TM2, ainsi que l'interfaçage de l'ensemble de ces modules. Le langage de programmation choisit nous à permis de côtoyer le matériel de manière à ce qu'elle sera impossible à rivaliser avec un autre langage, pour approfondir nous connaissance sur le fonctionnement interne de ces modules.

Grace aux connaissances acquises dans ce mémoire, et l'ascension fulgurante auquel fait face notre pays dans le domaine de l'embarqué, nous ouvre des perspectives réel pour des débauché sur le marché existant.