

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE
UNIVERSITE MOULOUD MAMMARI TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE**

MEMOIRE

De fin d'étude

En vue de l'obtention du Diplôme de MASTER Académique
en Electronique
Option : Télécommunications et Réseaux

Thème

**Conception et réalisation d'un système de
vidéosurveillance
A base de la carte ARDUINO UNO**

Dirigé par :

M^r BENNAMENE.K

Réalisé par :

M^r BEN LOUNES Said

M^r AHNOUCHE Djamel

Promotion 2013/2014

Remerciements

Nous remercions le Bon Dieu de nous avoir donné la force, le courage et la volonté pour accomplir ce travail.

Nous remercions notre promoteur Mr. BENNAMANE pour son encadrement et son aide ainsi que ses conseils tout au long de ce projet.

Nous remercions également le président de jurys ainsi que les membres de jurys d'avoir accepté d'évaluer et de juger notre travail.

Nous remercions aussi pour les gens qui nous ont aidé toute au long de ce travail : M^r BERCHICHE, M^{elle} BELHADJ, M^r NACHEF, M^r TOUMI, M^r ZERROUKI, M^r AIT MOUKHTAR MAHDI, M^r KHELFOUN.

Enfin, un remerciement tout particulier à toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce projet.

Dédicaces

Nous remercions ALLAH tout puissant de nous avoir donné la force et le courage afin d'accomplir ce travail

Je dédie ce travail :

A mon cher père et ma chère mère, mes frères JUGURTHA et SOFIANE , mes belles sœurs et leur petits anges « ZAKARIA » et « RANIA », ma sœur « HANANE » , qui m'apportent leur soutien moral et leur force à chaque moment de ma vie ;

A Mon binôme BEN LOUNES Said avec qui j'ai passé des agréables moments ainsi que sa famille ;

A « OUCHAOUA Fatiha » qui était toujours derrière mes succès ;

A mes meilleurs amis « CHERTOUH Amar » et « IKENE Samir » ;

Ainsi à tous mes amis qui ont marqué un souvenir et étaient contents pour ma réussite

Djamel

Dédicaces

Je dédie ce modeste travail :

A ma trèschère mère et à mon père qui ont beaucoup sacrifié avant de voir le fruit de leur éducation ;

A mon frère HACENE ;

A ma sœur OUARDA, son mari SMAIN et leurs enfant RAYAN et la petite princesse YASMINE ;

A ma sœur KAHINA, son mari AMAR ;

Qui m'ont soutenu toute au long de ma vie.

A mon binôme DJAMEL et à toute sa famille ;

A mon ami BELAID ;

A tous mes ami(e)s, sur tous mes copains de chambre.

SAID

SOMMAIRE

Introduction.....	3
-------------------	---

CHAPITRE I :

1. Evolution des systèmes de vidéosurveillance	5
1.1. Systèmes de vidéosurveillance classique CCTV	5
1.2. Systèmes de vidéosurveillance analogique avec magnétoscopes traditionnels.....	5
1.3. systèmes de vidéosurveillance analogique avec enregistreurs numériques	6
1.4. Systèmes de vidéosurveillance analogique avec enregistreurs numériques réseau.....	7
1.5. Systèmes de vidéo sur IP avec serveurs vidéo	7
1.6. Systèmes de vidéo sur IP avec cameras réseau	9
1.7. Réseau sans fil	10
2. Architecture d'une installation de Vidéosurveillance.....	11
2.1. Vidéosurveillance en circuit fermé ou CCTV	11
2.2. Vidéosurveillance en circuit ouvert ou OCCTV	12
3. Fonctions d'une installation télésurveillance	12
3.1. Fonction réception	13
3.2. Fonction gestion	13
3.3. Fonction visualisation	13
4. Technologies des caméras.....	13
4.1. Caméra à capteur CCD (Charge CoupledDevice).....	14
4.2. Caméra CMOS	15
5. Les principaux modèles de cameras	15
5.1. Caméra analogique	15
5.2. Caméra discrète	16
5.3. Caméra numérique (caméra IP).....	16
5.4. Caméras jour / nuit.....	17
5.5. Caméras infrarouges (IR).....	17
5.6. Caméras espions ou mini caméras	18
5.7. Dômes motorisés.....	18
5.8. Webcam vidéosurveillance	19

CHPITRE II :

1. Proposition d'un système de télésurveillance	20
1.1. principe de fonctionnement du système proposé.....	20
1.2. Matériels et logiciels utilisés	20
1.2.1. Carte Arduino UNO.....	20
1.2.1.1. Description de la carte ARDUINO UNO.....	20
1.2.1.2. Alimentation de la carte ArduinUNO.....	21
1.2.1.3. Protection du port USB contre la surcharge en intensité.....	22
1.2.1.4. Gestion des memoires dans la carte ARDUINO.....	22
1.2.1.5. Les entrés/sorties numériques.....	23
1.2.1.6. Les entrées analogiques.....	24
1.2.1.7. ARDUINO et la communication avec l'extérieur.....	25
1.2.1.8. Dimension de la carte.....	25
1.2.1.9. Le microcontrôleur ATmega328.....	26

SOMMAIRE

1.2.1.10.	Les principales caractéristiques de l'ATMega328.....	26
1.2.2.	L'environnement de programmation ARDUINO.....	28
1.2.2.1.	Description du logiciel ARDUINO.....	29
1.2.2.2.	Description de la barre des boutons.....	31
1.2.2.3.	Description de la barre des menus.....	32
1.2.2.4.	Les étapes de téléversement d'un programme vers la carte ARDUINO.....	33
1.2.3.	La carte Eye-Fi.....	33
1.2.3.1.	Eye-Fi Connect X2 4Go.....	33
1.2.3.2.	Caractéristiques techniques de l'Eye-Fi Connect X2.....	34
1.2.3.3.	Description du logiciel.....	35
1.2.3.4.	Paramétrage de la carte Eye-Fi.....	36
1.2.3.4.1.	Les Wifi réseaux privés et réseaux publics.....	36
1.2.3.4.2.	Le Wifi Direct-Mode.....	37
1.2.3.5.	Sélection et gestion des formats transférés.....	38
1.2.3.6.	Principe de fonctionnement d'Eye-Fi.....	40
1.2.4.	Shield carte SD.....	42
1.2.4.1.	Description du shield carte SD.....	42
1.2.4.2.	Caractéristiques.....	42
1.2.4.3.	Les principaux composants du shield.....	42
1.2.5.	Camera série TTL.....	44
1.2.5.1.	Description de la camera série TTL.....	44
1.2.5.2.	Câblage de l'appareil photo.....	45
1.2.5.3.	Utilisation de l'appareil.....	46
1.2.5.4.	Détection de mouvement.....	46
1.3.	Tests et résultats obtenus.....	47
1.3.1.	Tests de captures.....	47
1.3.1.1.	Résolution.....	47
1.3.1.2.	Vitesse de déplacement de la cible	47
1.3.2.	Tests de transmission.....	48
1.3.2.1.	Transmission vers un compte(type facebook).....	48
1.3.2.2.	Transmission vers un micro ordinateur.....	49
1.3.2.3.	Transmission vers un Smartphone.....	50
1.3.3.	Procédure de transmission	51
	Conclusion.....	52
	Bibliographie	
	Annexes	

SOMMAIRE

Liste des figures

Chapitre I

Figure 1: Vidéosurveillance en circuit fermé ou CCTV.....	5
Figure 2: système de vidéosurveillance analogique utilisant un magnétoscope.....	6
Figure 3: système de vidéosurveillance analogique utilisant un enregistreur numérique.....	6
Figure 4: Systèmes de vidéosurveillance analogique avec enregistreurs numériques réseau.....	7
Figure 5: Systèmes de vidéo avec serveurs vidéo.....	8
Figure 6: Systèmes de vidéo sur IP avec serveurs vidéo.....	8
Figure 7: Systèmes de vidéo sur IP avec cameras réseau.....	9
Figure 8: système vidéosurveillance utilise les deux technologies filaire et sans file.....	11
Figure 9: vidéosurveillance en circuit fermé ou CCTV.....	12
Figure 10: Synoptique d'une installation de télésurveillance.....	12
Figure 11: camera à capteur CCD.....	14
Figure 12: capteur CCD.....	14
Figure 13: Camera a capteur CMOS.....	15
Figure 14: Capteur CMOS.....	15
Figure 15: Caméra analogique.....	16
Figure 16: caméras discrètes.....	16
Figure 17: caméra IP.....	17
Figure 18: Cameras jour/ nuit.....	17
Figure 19: cameras infrarouges.....	18
Figure 20: caméras espions.....	18
Figure 21: dôme motorisé.....	18

Chapitre II

Figure 22: Carte ARDUINO UNO.....	21
Figure 23: Brochage de la carte ARDUINO UNO.....	24
Figure 24: le microcontrôleur ATmega328.....	26
Figure 25: Architecture interne de l'ATmega328.....	28
Figure 26: Fenêtre principale de l'environnement de programmation ARDUINO.....	29
Figure 27: Les différentes parties de la fenetre principale du logiciel Arduino.....	30
Figure 28: Fenêtre associée au moniteur série.....	31
Figure 29: structure de la barre des boutons.....	31
Figure 30: la barre des menus.....	32
Figure 31: Eye-Fi Connect X2 4Go.....	33
Figure 32: PC annonce qu'une nouvelle carte Eye-fi a été découverte.....	35
Figure 33: Application Eye-fi.....	36
Figure 34: Paramétrage wifi des réseaux privés : Ouvert, Wep, WPA.....	36
Figure 35: autorisation de transfère via un réseau.....	37
Figure 36: Paramétrage du Wifi en Direct Mode.....	37
Figure 37: Réseau Ad-hoc créé par la carte Eye-Fi.....	38
Figure 38: Organisation du dépôt des photos sur l'ordinateur.....	39
Figure 39: Publication sur internet.....	39
Figure 40: vu de face d'un shield SD carte.....	42
Figure 41: les composants du shield.....	43
Figure 42: camera serie TTL.....	44
Figure 43: camera série TTL.....	45
Figure 44: camera série TTL avec sortie vidéos.....	46

SOMMAIRE

Figure 24: Image 640X480	47
Figure 45: Image 320X240	47
Figure 46: Image 160X120	47
Figure 47: Personne en déplacement	47
Figure 48: Objet en chute libre.....	47
Figure 49: images de notre teste après transmission vers facebook.....	48
Figure 50: dossiers de sauvegardes	49
Figure 51: contenues des sauvegardes.....	49
Figure 52: application Eye Fi sous un Smartphone.....	50
Figure 53: images transmissent vers un Smartphone	50

Introduction

De nos jours, la vidéosurveillance est la solution la plus performante en matière de sécurité. On la retrouve dans de nombreux secteurs d'activités (banque, transports, industrie, grande distribution, etc.) et les différents sites de vie (villes, immeubles, base de vie, etc.).

Un système de vidéosurveillance assurera la protection des habitations, des entreprises, ou encore des commerces. On peut en quelques cliques, relire les vidéos et visualiser les caméras en direct. Les caméras de surveillance, équipées de la fonction détection de mouvement, jouent aussi le rôle d'alarme. Si la caméra détecte une présence, elle commence automatiquement à enregistrer, et peut aussi envoyer des alertes pare-mail.

Un autre avantage des caméras de surveillance, c'est la vision à distance. Il suffit de se connecter via un ordinateur ou un Smartphone pour visualiser ce qu'il se passe en direct.

La vidéosurveillance a commencé au début des années 1950. Les installations ne comportaient que des caméras et des écrans. Ensuite est apparu le matériel d'enregistrement permettant un archivage des scènes enregistrées.

Une installation de vidéosurveillance comprendra toujours tout ou une partie des équipements suivants :

- caméras,
- moniteurs,
- équipements de sauvegarde,
- équipements de gestion,
- la connectique.

Dans ce mémoire de master, on se propose de concevoir et de réaliser un système de vidéosurveillance avec détection de mouvement et transmission

de l'information vers un équipement ou un site distants (Smartphone, PC, site web, etc). Pour cela on a organisé le rapport en deux parties, une première partie sur l'état de l'art des systèmes de vidéosurveillance, ou les différents systèmes et configurations sont passés en revue. La deuxième partie du rapport est consacrée à la présentation de notre propre système de vidéosurveillance, son principe de fonctionnement, matériels et logiciels utilisés et les résultats des différents testes.

Bien sur on terminera notre travail par une conclusion et des perspectives futures.

1. Evolution des systèmes de vidéosurveillance :

Les systèmes de vidéosurveillance existent depuis environ 60 ans. Intégralement analogiques à leurs débuts, ils ont évolué progressivement vers la technologie numérique. Les systèmes actuels ne ressemblent plus guère aux anciennes caméras analogiques branchées sur des magnétoscopes traditionnels. Aujourd'hui, ils utilisent les caméras réseau et les serveurs informatiques pour l'enregistrement vidéo dans un système entièrement numérique.

1.1. Systèmes de vidéosurveillance classique CCTV :

Un système de vidéosurveillance classique CCTV (closed-circuit television) utilisant des caméras analogiques avec sorties coaxiales sont reliées au moniteur. Un opérateur doit être toujours présent devant la télé pour le contrôle et l'intervention aux cas anormaux.

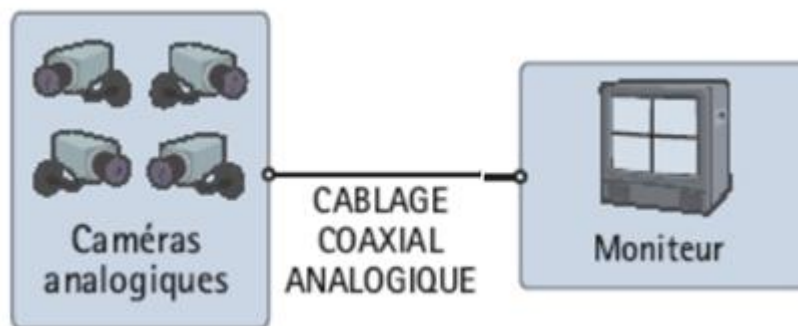


Figure 1: Vidéosurveillance en circuit fermé ou CCTV

1.2. Systèmes de vidéosurveillance analogique avec magnétoscopes traditionnels :

Un système de vidéosurveillance analogique utilisant un magnétoscope traditionnel VCR (Video Cassette Recording) est un système entièrement analogique dans lequel les caméras analogiques avec sorties coaxiales sont reliées au magnétoscope pour l'enregistrement, en utilisant un quad/multiplexeur qui permet alors d'enregistrer le contenu de plusieurs caméras sur un même magnétoscope. La surveillance vidéo nécessite un moniteur analogique.

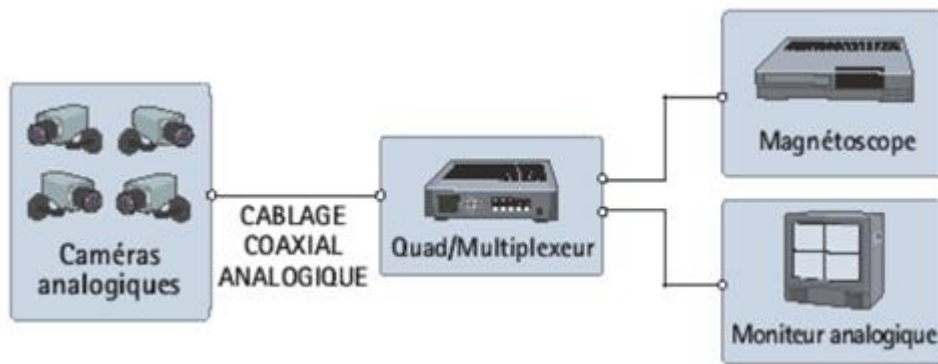


Figure 2: système de vidéosurveillance analogique utilisant un magnétoscope

1.3. systèmes de vidéosurveillance analogique avec enregistreurs numériques :

Un système de vidéosurveillance analogique utilisant un enregistreur numérique DVR (Digital Video Recorder) est un système analogique permettant l'enregistrement numérique des images. Avec un enregistreur numérique, l'enregistrement vidéo ne se fait plus sur bandes magnétiques mais sur des disques durs où les séquences sont numérisées et compressées de manière à emmagasiner chaque jour un maximum d'images. Les premiers enregistreurs numériques disposaient d'un espace disque limité. La durée des enregistrements était donc assez restreinte, à moins de réduire la fréquence d'images. Grâce aux progrès récents dans ce domaine, l'espace disque ne pose plus réellement problème. La plupart des enregistreurs numériques disposent en outre de plusieurs entrées vidéo (en général 4, 9 ou 16), ce qui leur permet d'intégrer d'emblée les fonctionnalités du quad ou des multiplexeurs.

Les enregistreurs numériques présentent les avantages suivants :

- Pas besoin de changer de cassette
- Qualité constante de l'image

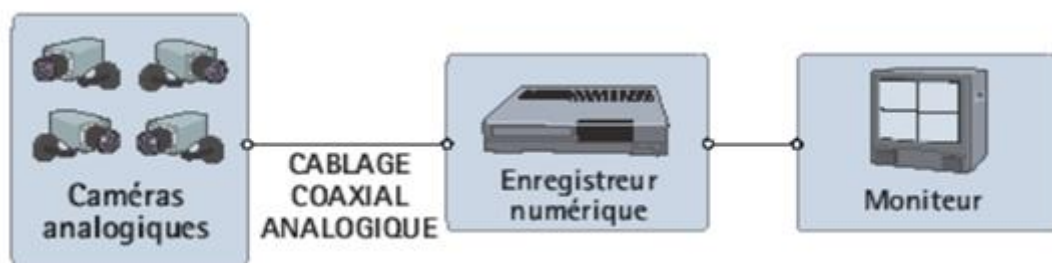


Figure 3: système de vidéosurveillance analogique utilisant un enregistreur numérique

1.4. Systèmes de vidéosurveillance analogique avec enregistreurs numériques réseau :

Un système de vidéosurveillance analogique passant par un enregistreur numérique (DVR) réseau est un système en partie numérique comprenant un enregistreur numérique réseau connecté via un port Ethernet. La vidéo étant numérisée et compressée sur l'enregistreur numérique, les images peuvent être transportées sur un réseau informatique à des fins de surveillance sur PC distant. Certains systèmes permettent à la fois la visualisation des séquences en direct et des séquences enregistrées; d'autres se limitent aux images enregistrées. Sur certains systèmes, la surveillance vidéo requiert en outre un client Windows spécifique, tandis que d'autres nécessitent un simple navigateur web standard, plus flexible pour une visualisation à distance.

Les enregistreurs numériques réseaux présentent les avantages suivants :

- Visualisation vidéo à distance sur PC
- Contrôle du système à distance



Figure 4: Systèmes de vidéosurveillance analogique avec enregistreurs numériques réseau

1.5. Systèmes de vidéo sur IP avec serveurs vidéo :

Un système de vidéo sur IP associé à un serveur vidéo comprend un serveur vidéo, un commutateur réseau et un PC équipé d'outils de gestion vidéo. La caméra analogique est branchée sur le serveur vidéo, lequel assure la numérisation et la compression des séquences vidéo. De son côté, le serveur vidéo est connecté sur le réseau qui transporte la vidéo vers un PC ou serveur via un commutateur réseau. La vidéo est alors enregistrée sur le disque dur du PC. Il s'agit alors d'un véritable système de vidéo sur IP. Ce système utilisant les serveurs vidéo présentent les avantages suivants :

CHAPITRE I

- Recours à un réseau standard et à un serveur informatique standard pour l'enregistrement et le traitement vidéo.
- Possibilité d'enregistrement hors-site.
- Système évolutif pouvant être élargi par ajout de nouvelles caméras réseau.

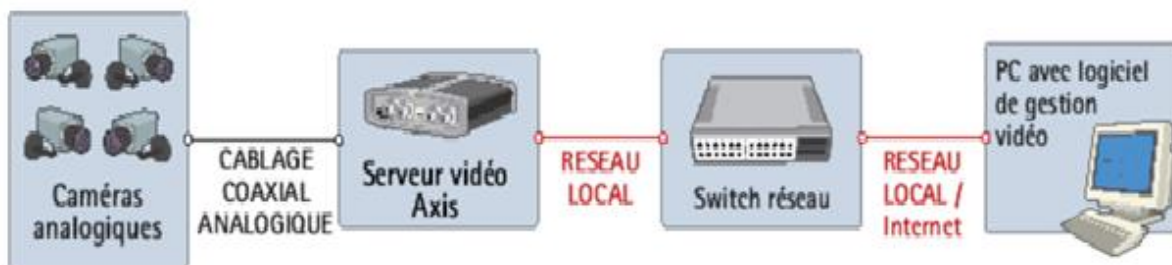


Figure 5: Systèmes de vidéo avec serveurs vidéo

Ce schéma présente un véritable système vidéo sur IP dans lequel les données vidéo sont transmises en permanence sur un réseau IP.

Le serveur vidéo sert de pierre angulaire pour assurer la migration du système de sécurité analogique vers une solution de vidéo sur IP.

Ø Serveurs vidéo

Les serveurs vidéo s'intègrent facilement dans un système existant de vidéosurveillance analogique CCTV (télévision en circuit fermé). Un serveur vidéo numérise les signaux vidéo analogiques et distribue des images numériques directement sur un réseau IP (par exemple, un réseau local LAN/intranet/Internet), en transformant les caméras analogiques en caméras réseau et en permettant aux utilisateurs de visualiser des images en direct à partir d'un navigateur web depuis tout ordinateur du réseau, en tout lieu et à tout moment.

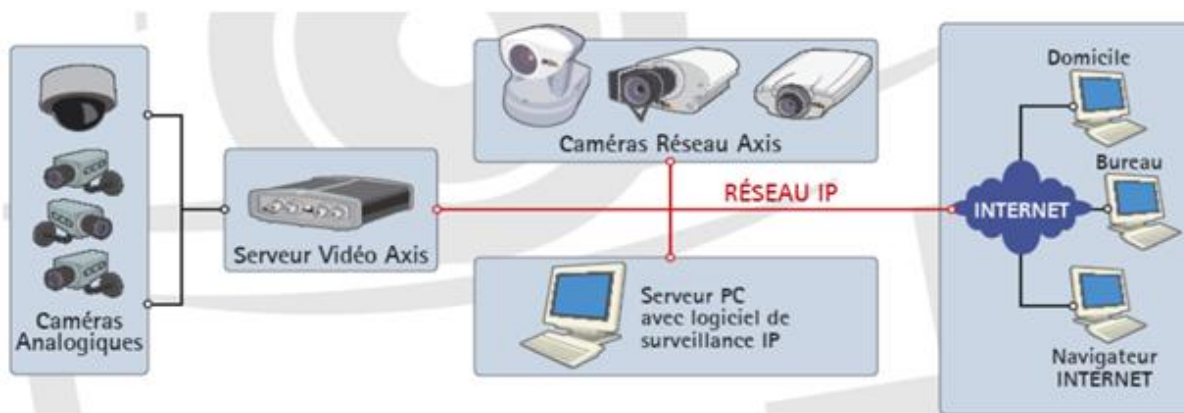


Figure 6: Systèmes de vidéo sur IP avec serveurs vidéo

CHAPITRE I

Un serveur vidéo comprend au moins une entrée vidéo analogique, un numériseur d'images, un compresseur d'images et un serveur web avec interface réseau. En associant un système de CCTV analogique à un système de vidéo réseau, vous disposez de toutes les fonctions et de tous les avantages de la technologie numérique :

Accès à distance, économie, souplesse, évolutivité, fonctions d'intégration et d'évolutivité et exceptionnelle qualité d'image.

1.6. Systèmes de vidéo sur IP avec cameras réseau :

Une caméra réseau associe une caméra et un ordinateur. Permettant la numérisation et la compression vidéo, elle est en outre équipée d'un connecteur réseau. La vidéo est acheminée par réseau IP via les commutateurs réseau, pour être enregistrée sur un PC/serveur standard à l'aide d'outils de gestion vidéo. Il s'agit d'un système de vidéo sur IP à part entière, doublé d'un système entièrement numérique n'utilisant aucun composant analogique.

Les systèmes de vidéo sur IP reposant sur l'utilisation de caméras réseau présentent les avantages suivants :

- Caméras haute résolution (méga pixels)
- Qualité constante de l'image
- Fonction d'alimentation par câble Ethernet (Power over Ethernet) et réseau sans fil
- Fonctions panoramique/inclinaison/zoom, audio, entrées et sorties numériques sur IP
- Grandes flexibilité et évolutivité

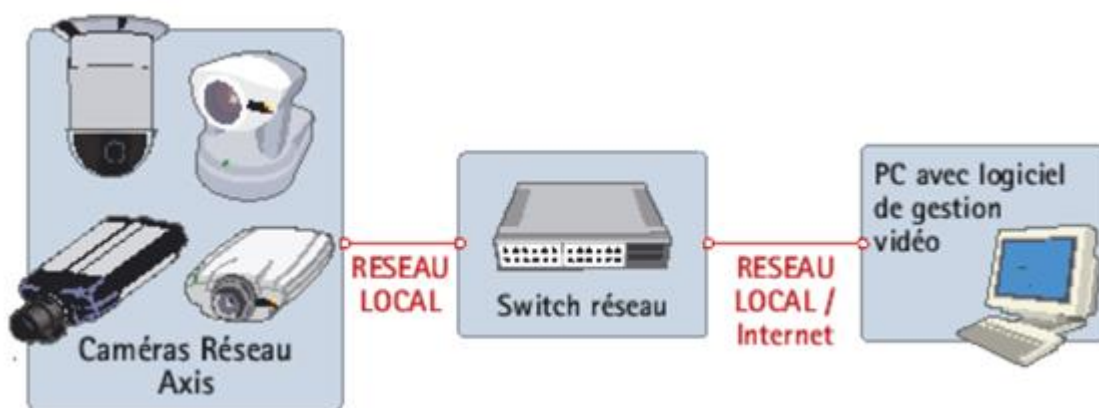


Figure 7: Systèmes de vidéo sur IP avec cameras réseau

Ce schéma présente un système de vidéo sur IP dans lequel la vidéo est transmise en permanence sur réseau IP à l'aide de caméras réseau. Ce système permet de profiter

CHAPITRE I

pleinement de tous les avantages offerts par la technologie numérique tout en offrant une qualité d'image constante entre la caméra et l'écran, quel que soit l'endroit où ils se trouvent.

1.7. Réseau sans fil :

Même si les réseaux filaires prévalent actuellement dans la plupart des bâtiments, une solution sans fil peut s'avérer intéressante pour l'utilisateur, tant financièrement que sur le plan fonctionnel. Songeons par exemple à certains bâtiments classés, où l'installation d'un câblage endommagerait inévitablement l'intérieur, ou à certains sites (commerces par exemple) pour lesquels la caméra doit être régulièrement déplacée et où l'on ne souhaite pas devoir tirer chaque fois de nouveaux câbles.

Une autre utilisation courante de la technologie sans fil concerne les bâtiments ou les sites que l'on souhaite relier sans pour autant devoir entreprendre de lourds et coûteux travaux au sol.

La technologie sans fil s'applique à la fois aux systèmes de vidéo sur IP et aux systèmes analogiques.

Elle dépasse donc le périmètre strict des réseaux. La transmission sans fil se divise en deux catégories principales :

LAN sans fil (Wireless LAN, ou WLAN) :

Le LAN définit un réseau local, c'est-à-dire sur de courtes distances et en principe à l'intérieur.

Les normes LAN sont aujourd'hui bien définies et les périphériques de marques différentes sont généralement compatibles entre eux.

Ponts sans fil

Lorsque certains bâtiments ou sites doivent être reliés par une liaison rapide, une liaison point à point longue distance et à grande vitesse est nécessaire. Les technologies micro-ondes et laser sont couramment utilisées.

Réseau typique avec connexion filaires et sans fils

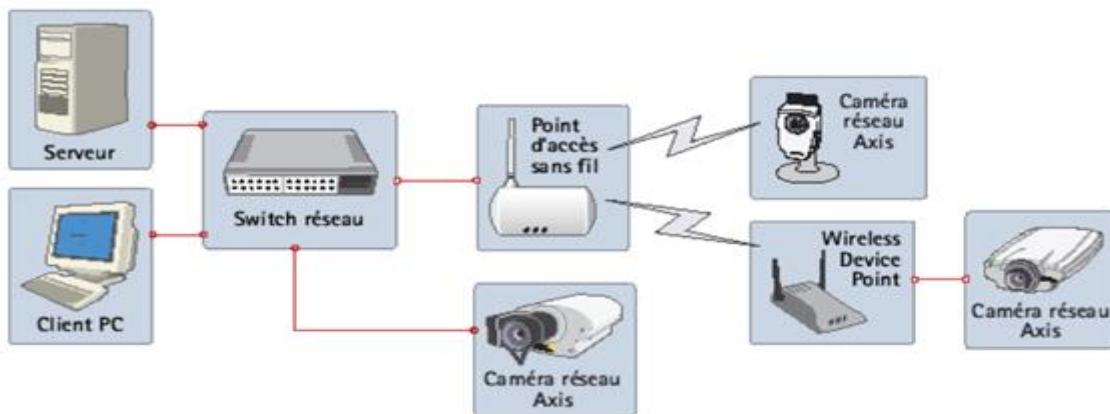


Figure 8: système vidéosurveillance utilise les deux technologies filaire et sans file

Ce schéma présente un usage combiné des deux technologies:

- Le point central est un Switch/commutateur réseau. A gauche, un serveur (en haut) et un client PC (en bas) sont connectés via un câblage Ethernet. A côté du Switch se trouve un point d'accès sans fil. Celui-ci gère tous les appareils sans fil à proximité.
- Deux appareils sans fil sont représentés dans ce schéma:
 - ü La Caméra Réseau AXIS 206W. Cette caméra intègre le support pour les communications sans fil.
 - ü Un Wireless Device Point. Cet appareil fournit la communication sans fil et se connecte directement au point d'accès sans fil.

Il est également possible de connecter le serveur et le PC sans fil. La bande passante en mode sans fil étant limitée par rapport aux communications filaires, il est préférable d'utiliser les réseaux câblés aussi souvent que possible.

2. Architecture d'une installation de Vidéosurveillance :

vidéosurveillance en circuit fermé et circuit ouvert:

2.1. Vidéosurveillance en circuit fermé ou CCTV :

Une installation de vidéosurveillance en circuit fermé (ou CCTV, Closed Circuit Television), le système est constitué d'un réseau de caméras et de moniteurs appartenant à une structure ou organisation n'ayant pas pour vocation de diffuser les images hors de ses murs. L'émission et la réception n'intéressent que celui qui est relié au réseau.

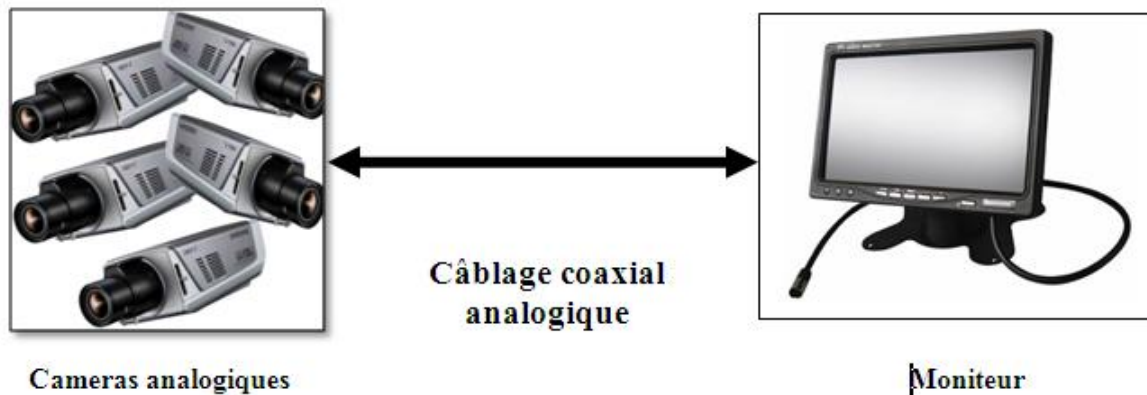


Figure 9:vidéosurveillance en circuit fermé ou CCTV

2.2. Vidéosurveillance en circuit ouvert ou OCCTV :

Le rôle d'un système de vidéosurveillance est d'accomplir une tâche bien définie de sécurité pour un site. Or il arrive que l'utilisateur souhaite une gestion multi site – il est tout à fait légitime qu'il puisse accéder à son système à distance, et cela en toute sécurité. Dans ce secteur, des progrès ont été accomplis grâce aux technologies électroniques, informatiques et télécoms. Le terme OCCTV (Open Closed Circuit Television) a été créé pour décrire ce type d'application.

3. Fonctions d'une installation télésurveillance :

Dans une installation de télésurveillance, il y a toujours trois fonctions importantes et interdépendantes : réception, gestion, visualisation.

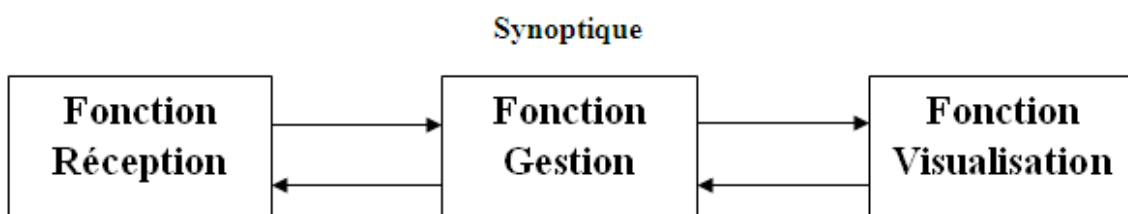


Figure 10:Synoptique d'une installation de télésurveillance.

Fonction réception: Cameras.

Fonction gestion: Commutateur, Multiplexeur, Magnétoscope, Enregistreur numérique, Ordinateur, Logiciels.

Fonction visualisation: Moniteurs

3.1.Fonction réception :

La caméra est l'élément fondamental du système de télésurveillance. En fonction de l'environnement et des besoins de l'utilisateur, il conviendra de choisir dans une gamme pléthorique de caméras le matériel adéquat :

- des caméras couleur ou noir et blanc.
- des caméras hautes définition.
- des caméras couleur commutable noir et blanc.
- des caméras fixes, mobiles, discrètes.
- des caméras intérieures ou extérieures.
- de l'environnement (éclairage, champ électrique, etc.

3.2.Fonction gestion :

Les équipements de gestion permettent, en fonction des besoins d'exploitation, d'afficher une ou plusieurs images sur un ou plusieurs écrans. Ce type d'affichage se fera au travers de différents matériels prédéfini lors de l'étude et correspondra aux besoins exprimés par le client.

3.3.Fonction visualisation :

Cette fonction est souvent agrégée au poste de garde ou au PC de sécurité. Elle se fait sur un moniteur qui est l'écran qui permet de visualiser les images capturées par les caméras de surveillances ou enregistrées par l'enregistreur de télésurveillance.

4. Technologies des caméras :

La caméra est un système de prise d'images animées qui génère un signal vidéo noir et blanc ou couleur. La caméra capte la lumière pour la transformer en signal électrique. Elle se compose parfois d'un capteur d'image et d'une électronique de traitement permettant de générer le signal vidéo. On peut trouver actuellement sur le marché plusieurs types de caméras qui peuvent répondre aux besoins d'un utilisateur selon ses objectifs. Pour bien choisir sa camera de surveillance, il est important de connaître ses performances. La qualité de l'image peut se mesurer à partir des critères comme:

CHAPITRE I

- selon la qualité de l'objectif, la lentille laisse passer plus ou moins de lumière.
- plus le focal est petit, plus l'angle de vue est large mais l'image est moins nette (pour une image nette, mieux vaut opter pour un petit angle de vue).
- plus ils sont sensibles à la lumière, meilleure est l'image lorsque la luminosité est faible.
- plus leur nombre de points perceptibles est important (pixels ou résolution), meilleure est également l'image.
- il existe des capteurs CMOS ou CCD.

4.1. Caméra à capteur CCD (Charge Coupled Device) :

Le capteur CCD ne fait pas de distinction entre les couleurs. L'analyse des couleurs se fait via des filtres qui permettent de récupérer les signaux RVB avant traitement.



Figure 11:camera à capteur CCD



Figure 12:capteur CCD

4.2. Caméra CMOS

La technologie CMOS est très utilisée dans la fabrication de composants électroniques, ce qui la rend économique pour la fabrication de caméras CMOS. Elles permettent une intégration à très grande échelle et consomment une quantité minimale d'énergie par rapport aux CCD ; une caméra peut fonctionner avec une batterie au NiCd pendant une semaine tandis que la caméra CCD ne fonctionnerait que quelques heures.

La caméra CMOS génère un signal comportant du bruit nuisant à la fourniture d'une image de qualité. Cependant des progrès sensibles se font sentir et la caméra CMOS gagne inexorablement du terrain sur sa consœur la caméra CCD.



Figure 13: Camera à capteur CMOS

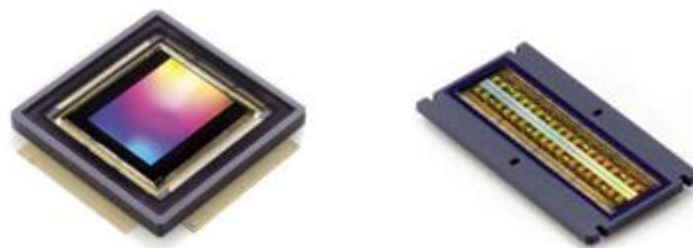


Figure 14: Capteur CMOS

5. Les principaux modèles de caméras :

5.1. Caméra analogique :

Ces caméras sont facilement reconnaissables ; elles ont une sortie de type BNC, la liaison se fait via le câble coaxial



Figure 15: Caméra analogique

5.2. Caméra discrète :

Les progrès des technologies font que de nos jours une caméra peut se loger dans n'importe quel accessoire. Les plus courants sont le détecteur d'intrusion, l'horloge et la tête de détection incendie



Figure 16: caméras discrètes

5.3. Caméra numérique (caméra IP) :

La camera de surveillance IP est utilisée pour des installations de surveillance en réseau, également appelées vidéosurveillance IP (Internet Protocol). La caméra IP est reliée à un serveur central (par liaison WiFi ou par câble RJ45) qui peut gérer plusieurs sites distants. Pour être reconnue sur le réseau et consultable depuis n'importe quel ordinateur, la caméra IP dispose de sa propre adresse IP. Ce système de vidéosurveillance présente de nombreux avantages, notamment pour les entreprises:

CHAPITRE I

- il dissuade les intrus et protège les locaux.
- les images capturées peuvent être visionnées en temps réel ou visionnées à distance à partir d'un ordinateur connecté au réseau (Internet, Intranet...) .
- il peut être paramétré en fonction des besoins: enregistrement vidéo 24h/24 et 7j/7 ou bien enregistrement uniquement lorsque l'alarme se déclenche, prise de photos, alerte par e-mail ou SMS (sur PC ou PDA), association avec un système d'alarme ou de détection de mouvement, d'intrusion, de chaleur.

Le seul inconvénient: l'installation d'une camera surveillance IP nécessite de bonnes connaissances en informatique pour le paramétrage.



Figure 17:caméra IP

5.4.Caméras jour / nuit :

Les caméras jour/nuit sont aussi performantes de jour comme de nuit. Les caméras jour/nuit identifient la luminosité de la zone filmée et basculent automatiquement en mode jour ou en mode nuit en fonction du niveau de luminosité détecté.



Figure 18:Cameras jour/ nuit

5.5. Caméras infrarouges (IR) :

Ce que l'homme ne peut pas distinguer dans l'obscurité, la caméra IR le voit ! Grâce aux LED qui entourent l'objectif de la caméra IR, vous bénéficierez d'une parfaite retransmission d'image, jusqu'à 100 mètres, même dans les zones les plus obscures. Il est important de choisir sa caméra IR par rapport à la distance de projection des infrarouges et en fonction de l'angle de vision souhaité.



Figure 19: caméras infrarouges

5.6. Caméras espions ou mini caméras :

Répondant aux mêmes caractéristiques techniques que les caméras d'intérieur, les caméras espions sont suffisamment petites pour passer inaperçues ou pour être dissimulées dans des objets.



Figure 20: caméras espions

5.7. Dômes motorisés :

Les dômes motorisés peuvent être installés sur un plafond ou un mur et sont pilotables grâce à un joystick ou une souris. Les dômes motorisés peuvent effectuer des rotations à 360°. Ils peuvent également zoomer sur les zones qui préoccupent. On peut aussi les programmer pour effectuer des rondes.



Figure 21:dôme motorisé

5.8.Webcam vidéosurveillance :

La webcam vidéosurveillance est une petite caméra connectée à un ordinateur: elle peut être intégrée au PC ou ajoutée par son utilisateur. Elle affiche en direct sur l'ordinateur les images capturées. La webcam vidéosurveillance permet deux types de vidéosurveillance :

- la surveillance d'un espace situé à proximité de l'ordinateur: on peut par exemple surveiller un bébé qui dort dans sa chambre pendant que l'on travaille sur un ordinateur dans une autre pièce ;
- la vidéosurveillance à distance:

En installant sur l'ordinateur un logiciel de vidéosurveillance (traitement des images), on peut visualiser en temps réel et de n'importe quel point du globe ayant accès à une connexion Internet, le lieu qui est mis sous surveillance. Le logiciel enregistre les images et permet de revisualiser des séquences.

Les logiciels les plus perfectionnés proposent de nombreuses applications: alerte de l'utilisateur en cas de détection d'une anomalie (mouvement, fumée...) par e-mail, SMS ou message téléphonique, alarme, vidéo en continu...

Il est possible de connecter plusieurs webcams sur un même ordinateur.

1. Proposition d'un système de télésurveillance :

1.1.principe de fonctionnement du système proposé :

Notre travail repose sur quatre modules électroniques (pièces) qui sont une caméra série de type TTL avec détecteur de mouvement intégré une carte mémoire de type SD Eye-fi avec option WIFI intégré, un shield comme support pour placer la camera et la carte mémoire.

Lorsque un mouvement est détecté avec une caméra équipée d'un détecteur de mouvement, une image sera prise et enregistrée dans une carte mémoire, puis transmise au réseau via ce support WIFI a un PC ou un Smartphone.

1.2.Matériels et logiciels utilisés :

1.2.1. Carte Arduino UNO :

1.2.1.1. Description de la carte ARDUINO UNO :

Modèle UNO de la société ARDUINO est une carte électronique dont le noyau est un microcontrôleur ATMEL de référence ATmega328. Le microcontrôleur ATmega328 est un microcontrôleur 8 bits de la famille AVR dont la programmation peut être réalisée en langage C. Cette carte possède 14 entrées/sorties numériques (dont 6 peuvent être utilisées comme étant des sorties **PWM (Pulse Width modulation)**), 6 entrées analogiques avec un convertisseur Analogique/Numérique de 10 bits de résolution, 1 résonateur céramique (quartz) de 16 MHz, 1 connecteur **ICSP (In Circuit Serial Programming)** qui permet d'injecter le bootloader à l'intérieur du microcontrôleur, 1 connecteur jack pour une alimentation extérieur, un bouton de reset pour mettre le processus à zéro.

L'avantage de cette carte c'est qu'elle n'a pas besoin de pilote pour faire la conversion FTDI USB/série, elle a juste un petit microcontrôleur **ATmega328** (pour la version 2) programmé comme convertisseur **USB/série**.

Elle contient tout ce que le microcontrôleur a besoin pour fonctionner, il faut seulement la connecter avec un câble USB à un ordinateur ou avec une alimentation externe.

L'intérêt principal des cartes ARDUINO est leur facilité de mise en œuvre. ARDUINO fournit un environnement de développement s'appuyant sur des outils open-source. Le chargement du programme dans la mémoire du microcontrôleur se fait de façon très simple par port USB. En outre, des bibliothèques de fonctions «clé en main» sont également fournies

CHAPITRE II

pour l'exploitation d'entrées-sorties courantes: gestion des E/S, gestion des convertisseurs ADC, génération de signaux PWM, exploitation de bus I2C, exploitation de servomoteurs, émission/réception série ...etc.

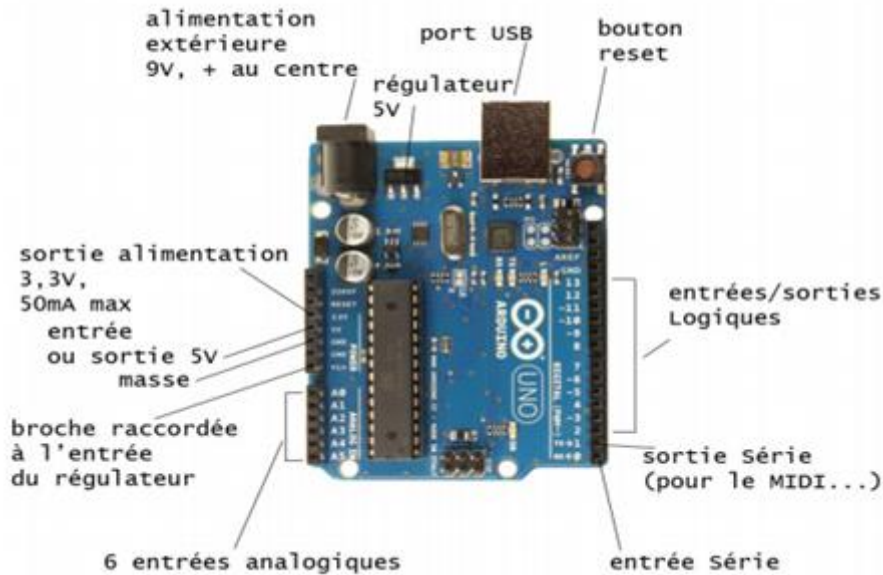


Figure 1: Carte ARDUINO UNO

1.2.1.2. Alimentation de la carte ArduinUNO :

La carte Arduino Uno peut être alimentée soit via la connexion USB (qui fournit 5V jusqu'à 500 mA) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte. L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12V sous 500 mA) ou des piles.

L'adaptateur secteur peut être connecté en branchant une prise de 2.1 mm dédiée du pôle positif au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs des broches de la carte appelées GND(masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation. La carte peut fonctionner avec une alimentation externe de 6 à 20 volts. Cependant, si la carte est alimentée avec moins de 7 V, la broche 5 V pourrait fournir moins de 5 V et la carte pourrait être instable. Si on utilise plus de 12 V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. Aussi, la plage idéale recommandée pour alimenter la carte Uno est entre 7V et 12V.

Il est à noter qu'il est strictement dangereux d'utiliser une alimentation externe via la prise jack et avoir le câble USB connecté (risque d'endommager de la carte).

CHAPITRE II

Les broches d'alimentation sont les suivantes:

- § **Vin**: (à distinguer du 5 V de la connexion USB ou autre source 5V régulée). On peut alimenter la carte à l'aide de cette broche, ou si l'alimentation est fournie par le jack d'alimentation, on peut accéder à la tension d'alimentation sur cette broche.
- § **5V**: la tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info: les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite «tension régulée» obtenue à l'aide d'un régulateur intégré dans la carte Arduino). Le 5V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation Vin via le régulateur de la carte, ou bien de la connexion USB (qui fournit du 5V régulé) ou de tout autre source d'alimentation régulée.
- § **3V3**: une alimentation de 3.3V fournie par le régulateur de 3,3V de la carte est disponible : ceci est intéressant pour certains circuits externes nécessitant cette tension au lieu du 5V). l'intensité maximale disponible sur cette broche est de 50 mA.

1.2.1.3. Protection du port USB contre la surcharge en intensité :

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

1.2.1.4. Gestion des memoires dans la carte ARDUINO :

L'ATmega328 a 32 Ko mémoire FLASH pour stocker le programme (dont 0.5Ko également utilisés par le bootloader). L'ATmega328 a également 2 Ko de mémoire SRAM (volatile) et 1 Ko d'EEPROM (non volatile – mémoire qui peut être lue à l'aide de **la librairie EEPROM**).

1.2.1.5. Les entrées/ sorties numériques :

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

Il y a entre ces broches celles qui ont des fonctionnalités en plus:

- **Communication série:** Broches 0 (RX) et 1 (TX). Utilisées pour recevoir (RX) et transmettre (TX) les données séries de niveau TTL. Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega328 programmé en convertisseur USB vers série de la carte (composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur). On fait appel à la transmission série à travers ces broches avec l'instruction **Serial.print()**, à condition que le câble USB soit déconnecté.
- **Interruptions Externes:** Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur.
- **Impulsion PWM (largeur d'impulsion modulée):** Broches 3,5,6,9,10 et 11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction **analogWrite()**.
- **SPI (Interface Série Périphérique):** broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la bibliothèque pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP.
- **I2C:** broches 4 (SDA) et 5 (SCL), supportent les communications de protocole I2C, disponible en utilisant la bibliothèque **Wire/I2C**.
- **LED:** broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.

1.2.1.6. Les entrées analogiques :

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

- **Autres broches:**

Il y a deux autres broches disponibles sur la carte:

- **AREF:** tension de référence pour les entrées analogiques (si différent du 5V), utilisée avec l'instruction **analogReference()** .
- **Reset:** mettre cette broche au niveau BAS entraine la réinitialisation (le redémarrage) du microcontrôleur. Typiquement, cette broche est utilisée pour ajouter un bouton de réinitialisation sur le circuit.

on peut voir les différentes broches de la carte et leurs fonctions à travers la figure suivante:

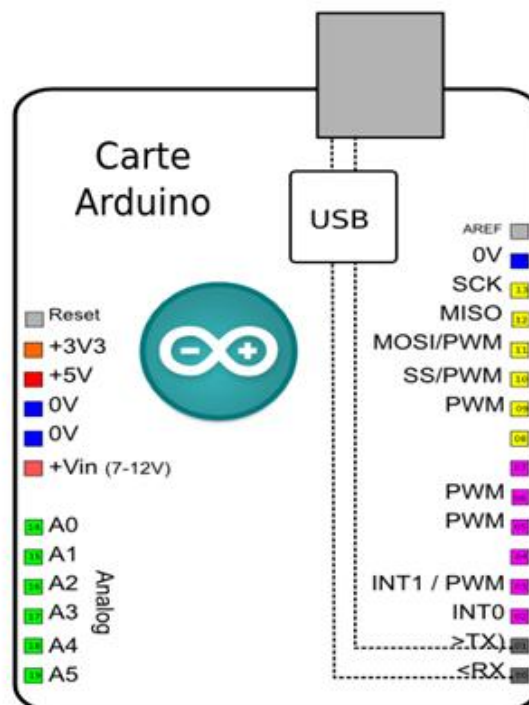


Figure 2:Brochage de la carte ARDUINO UNO

1.2.1.7. ARDUINO et la communication avec l'extérieur :

La carte Arduino Uno dispose de toute une série de facilités pour communiquer avec un ordinateur, une autre carte Arduino, ou avec d'autres microcontrôleurs.

L'ATmega328 dispose d'une UART (Universal Asynchronous Receiver Transmitter ou émetteur-récepteur universel asynchrone en français) pour communication série de niveau TTL (5V) et qui est disponible sur les broches 0 (RX) et 1 (TX).

Un circuit intégré ATmega8U2 sur la carte assure la connexion entre cette communication série vers le port USB de l'ordinateur et apparaît comme un port COM virtuel pour les logiciels de l'ordinateur. Le code utilisé pour programmer l'ATmega8U2 utilise le driver standard USB COM, et aucun autre driver externe n'est nécessaire.

Le logiciel Arduino inclut une fenêtre terminal série (ou moniteur série) sur l'ordinateur et qui permet d'envoyer des textes simples depuis et vers la carte Arduino. Les LEDs RX et TX sur la carte clignote lorsque les données sont transmises via le circuit intégré USB-vers-série et la connexion USB vers l'ordinateur (mais pas pour les communications série sur les broches 0 et 1). Une librairie Série logicielle permet également la communication série (limitée cependant) sur n'importe quelle broche numérique de la carte UNO.

- L'ATmega328 supporte également ces protocoles I2C et SPI:
- Le logiciel Arduino inclut la librairie **Wire** qui simplifie l'utilisation du bus I2C.
- Pour utiliser la communication SPI (Interface Série Périphérique), la librairie pour communication SPI est disponible, il suffit de la faire inclure dans le programme au niveau du logiciel Arduino lors de la programmation.

1.2.1.8. Dimension de la carte :

Les longueurs et largeurs maximales de la UNO sont respectivement 6.86 cm et 5.33 cm, avec le connecteur USB et le connecteur d'alimentation Jack s'étendant au-delà des dimensions de la carte. Quatre trous de vis permettent à la carte d'être fixée sur une surface ou dans un boîtier (pour l'embarquer sur un système). Noter que la distance entre les broches 7 et 8 est de 0.16 pouces, et 0.1 pouces séparant les autres broches.

CHAPITRE II

1.2.1.9. Le microcontrôleur ATmega328 :

Le microcontrôleur utilisé sur la carte ARDUINO UNO est un microcontrôleur ATmega328. C'est un microcontrôleur ATMEL de la famille AVR 8 bits.

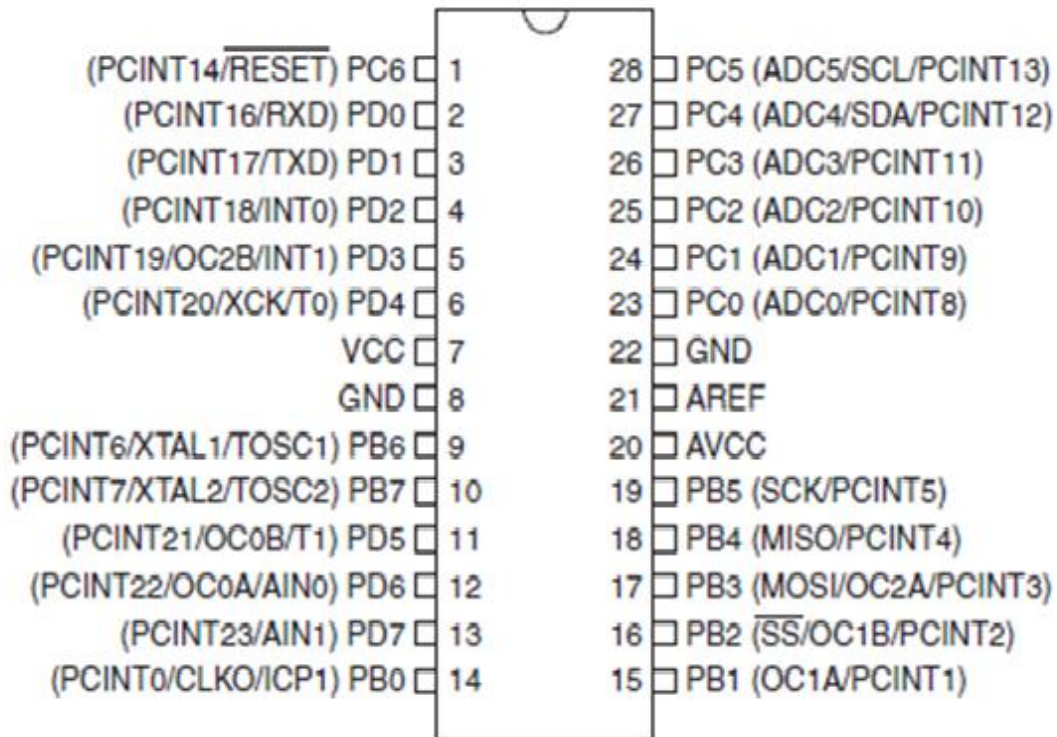


Figure 3:le microcontrôleur ATmega328

1.2.1.10. Les principales caractéristiques de l'ATmega328 :

Le microcontrôleur ATmega328 de Atmel dispose :

- De 14 broches numériques d'entrées/sorties, dont 6 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée), réparties selon l'ordre suivant : OC0A(PD6), OC0B(PD5), OC1A(PB1), OC1B(PB3), OC2A(PB3), OC2B(PD3) et 2 (0 et 1) pour réception/émission série).
- De 6 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques). Ces entrées /sorties sont réparties sur trois ports : Port B, Port D Port C (soit 23 broches E/S en tout).
- D'un courant max par broches E/S = 40 mA.
- D'un courant max sur sortie 3,3V= 50mA.
- D'une mémoire Flash de 32 KB dont 512B utilisée par le boot loader.

CHAPITRE II

- D'une mémoire SRAM de 2 KB.
- D'une mémoire EEPROM de 1KB.
- il contient aussi trois compteurs (Timer 0, Timer 1, Timer 2), le Timer 0 et le Timer 2 sont à comptage 8 bits, le Timer 1 il est à comptage 16 bits. Les Timers peuvent être utilisés pour générer les signaux PWM.

Certaines broches peuvent avoir plusieurs fonctions en fonction de ces programmations

PWM: Le ATmega a 6 broches qui peuvent servir à cette fonction qui sont les broches OC0A(PD6) , OC0B(PD5), OC1A(PB1), OC1B(PB3), OC2A(PB3), OC2B(PD3).

Convertisseur Analogique/Numérique: le ATmega328 possède un convertisseur Analogique/Numérique d'une résolution de 10 bits, ce convertisseur peut être utilisé à travers 6 entrées multiplexées de ADC0(PC0) jusqu'à ADC5(PC5).

Gestion du bus I2C: ce bus est exploité via les deux broches SDA(PC5)/SCL(PC4).

Port série (USART): émission/réception série via les broches TXD(PD1)/RXD(PD0).

Comparateur Analogique: le comparateur analogique intégré dans l'ATmega peut être utilisé à travers les deux broches AIN0(PD6) et AIN1(PD7), ce comparateur peut déclencher une interruption.

Watchdog Timer programmable: l'ATmega possède un compteur dit de **chien de garde** programmable pour générer des interruptions à la fin de son comptage et il peut être utilisé comme étant un simple compteur.

Gestion d'interruption (24 sources possibles) : en résumé

- Interruptions liées aux entrées INT0(PD2) et INT1(PD3).
- Interruptions sur changement d'état des broches PCINT0 à PCINT23.
- Interruptions liées aux Timers 0,1 et 2 .
- Interruption liée au comparateur analogique.
- Interruption de fin de conversion **ADC**.
- Interruptions du port série **USAR**.
- Interruption du bus **I2C**.

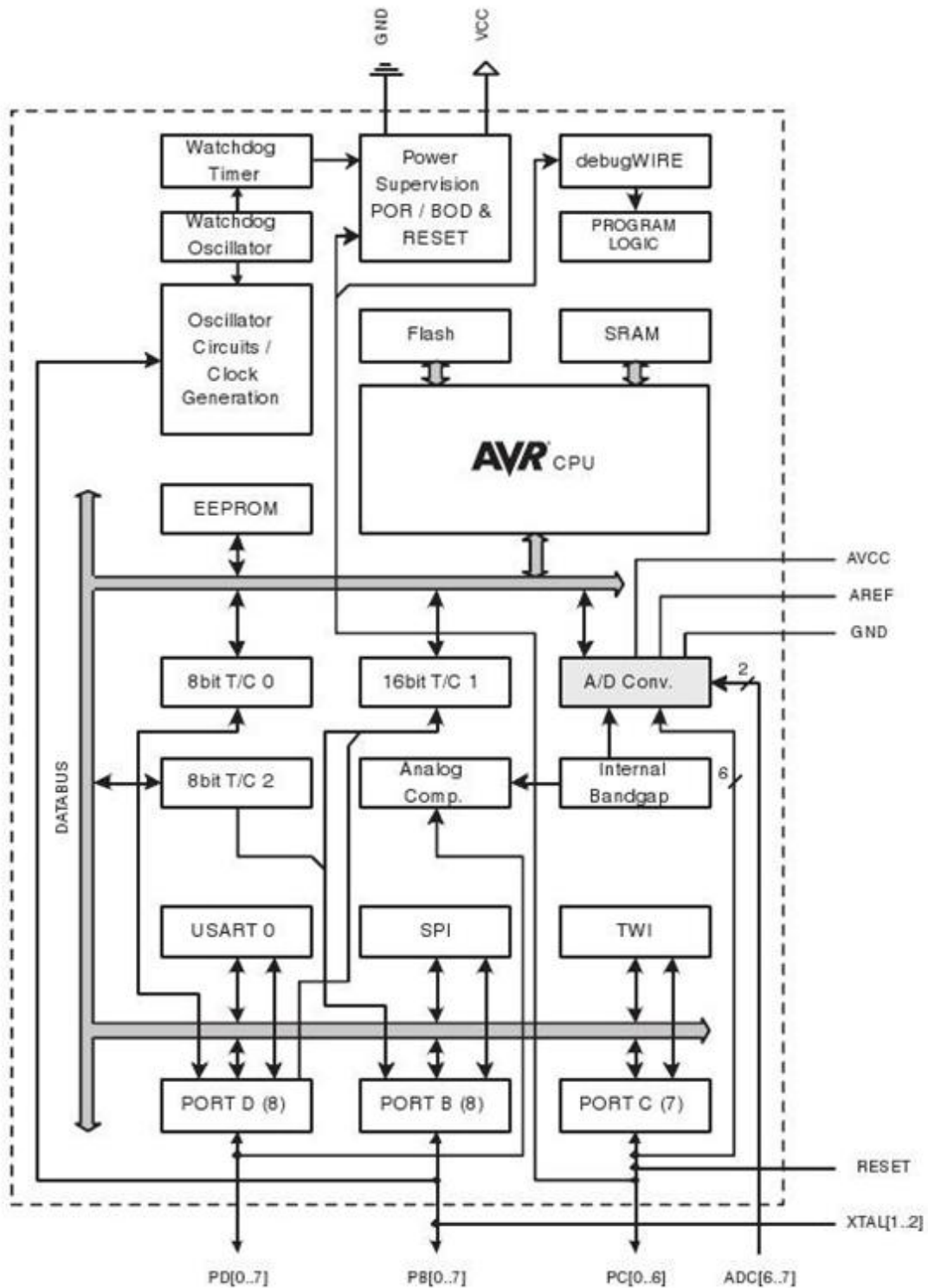


Figure 4: Architecture interne de l'ATmega328

1.2.2. L'environnement de programmation Arduino :

La carte Arduino présente le noyau de notre système, et pour qu'on puisse la programmer on doit disposer du logiciel compatible avec cette carte, ce logiciel est l'**Arduino EDI (Espace de Développement Intégré)**, qui porte le nom même de la carte Arduino.

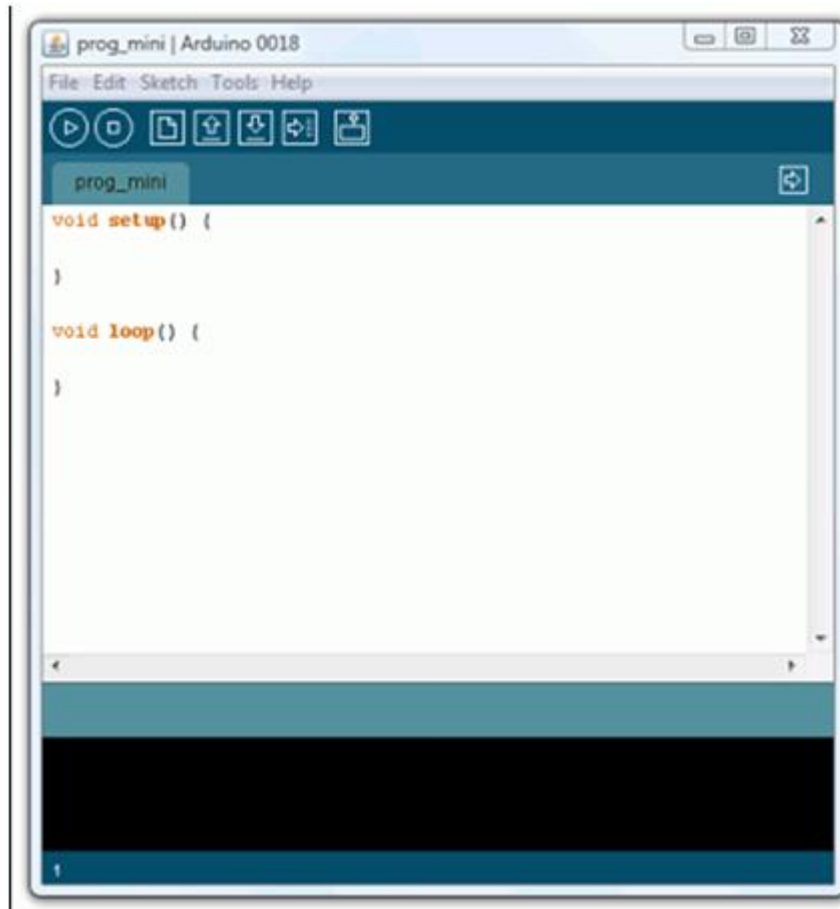


Figure 5: Fenêtre principale de l'environnement de programmation ARDUINO

1.2.2.1. Description du logiciel Arduino :

Le logiciel Arduino a pour fonctions principales:

- Le pouvoir d'écrire et compiler des programmes pour la carte Arduino.
- De se connecter avec la carte Arduino pour y transférer les programmes.
- De communiquer avec la carte Arduino.

Cet espace de développement intégré (EDI) dédié au langage Arduino et à la programmation des cartes Arduino.

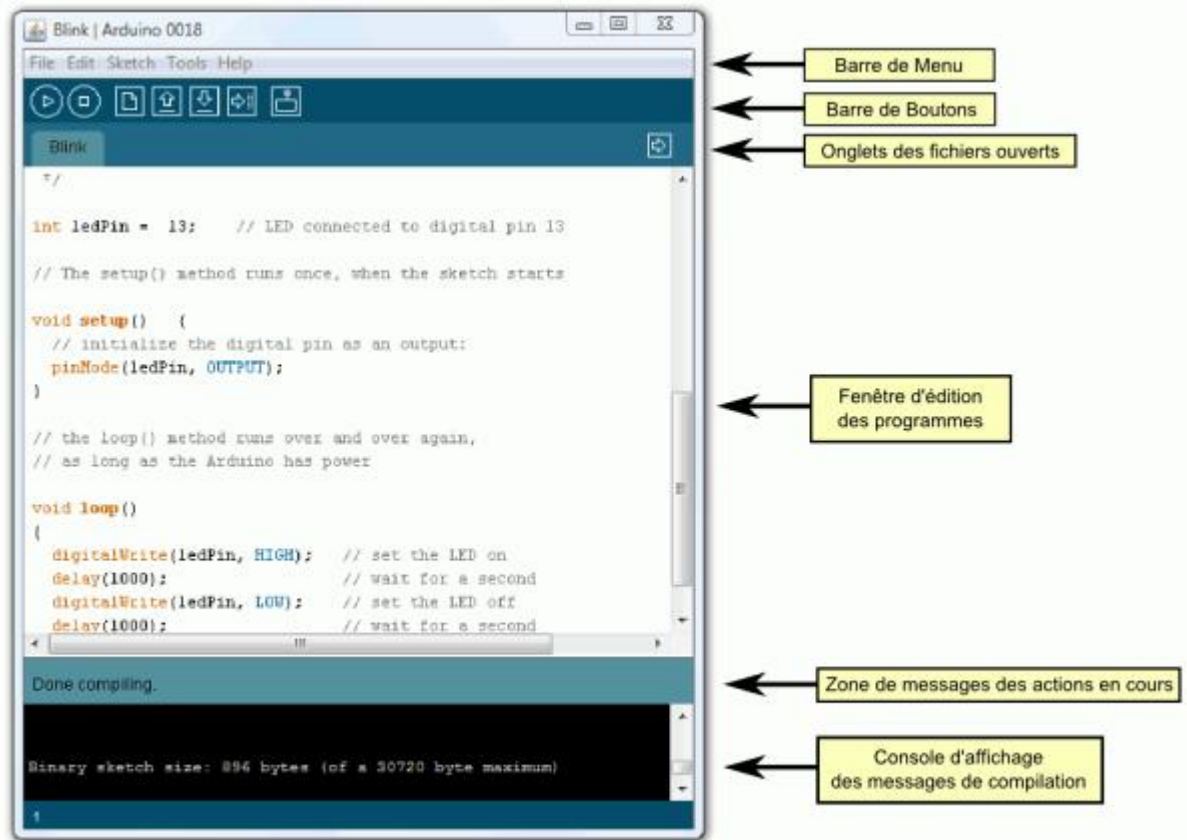


Figure 6: Les différentes parties de la fenêtre principale du logiciel Arduino

Il comporte:

- Une **BARRE DE MENUS**: comme pour tout logiciel une interface graphique (GUI).
- Une **BARRE DE BOUTONS**: cette barre donne un accès direct aux fonctions essentielles du logiciel.
- Un **EDITEUR** (à coloration syntaxique) : c'est dans cette espace ou on va écrire notre programme, il dispose aussi d'onglets de navigation.
- Une **ZONE DE MESSAGES**: permet d'afficher et d'indiquer l'état des actions en cours (ex : vérification, télé versement).
- Une **CONSOLE TEXTE**: elle permet d'afficher les messages concernant le résultat de la compilation du programme (il indique s'il y a des erreurs).
- **UNMONITEUR SERIE**: ce moniteur est utilisé pour la visualisation des données (message, valeurs, caractères) transmises vers l'ordinateur via le câble et d'envoyer des caractères vers la carte Arduino.

CHAPITRE II

- Cette fonctionnalité permet une mise au point facile des programmes et d'afficher sur l'ordinateur l'état des variables d'un programme ou il y a des calculs ou des mesures, aussi le résultat de la conversion analogique-numérique.

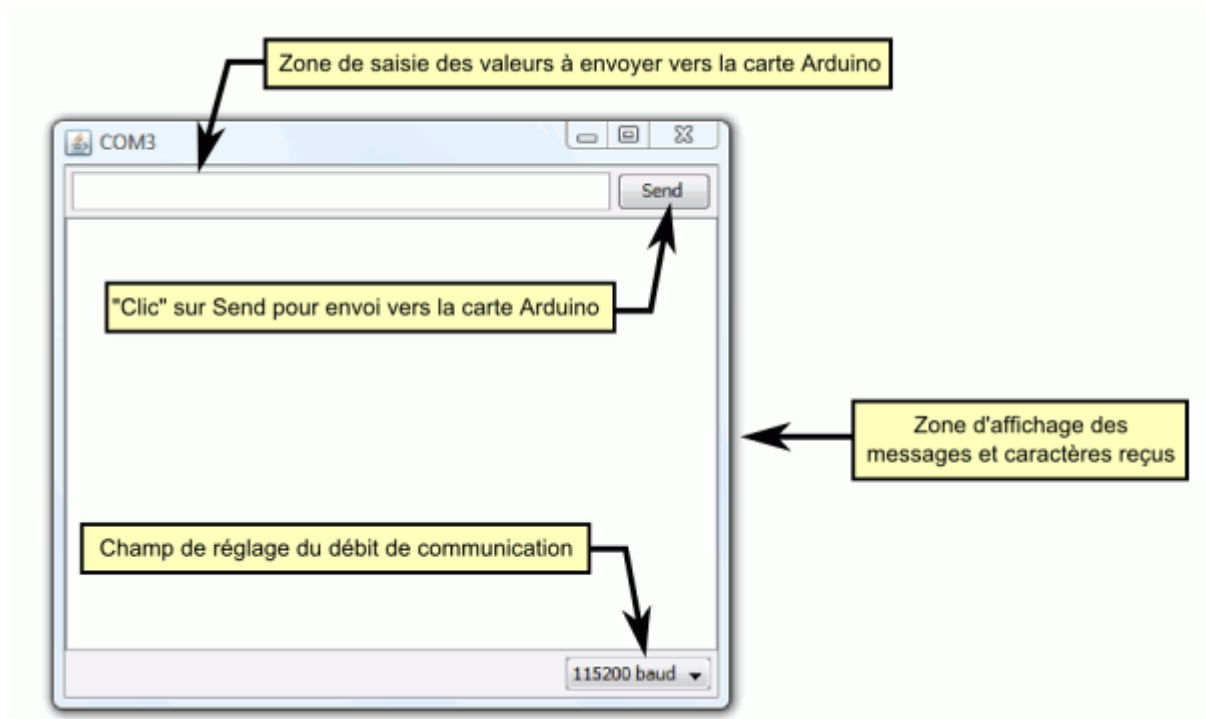


Figure 7: Fenêtre associée au moniteur série

1.2.2.2. Description de la barre des boutons :

La barre des boutons nous permet de vérifier la syntaxe, le transfert des programmes, aussi la création, l'ouverture et la sauvegarde d'un code écrit, elle permet aussi d'ouvrir le moniteur série. Elle dispose de sept boutons ou icônes :



Figure 8: structure de la barre des boutons



Vérifier/compiler: ce bouton nous permet de vérifier s'il y a des erreurs dans le programme en sélectionnant la ligne où se trouve l'erreur



Stop: ce bouton oblige le moniteur série ou les autres boutons activés de s'arrêter



Nouveau: en appuyant sur cette icône, une fenêtre d'édition vide s'ouvre pour la saisie d'un nouveau programme.



Ouvrir: il suffit de cliquer sur ce bouton pour accéder à la liste de tous les programmes qui se trouvent dans le «livre de programmes». Cliquer sur l'un des programmes l'ouvre dans la fenêtre courante.



Sauvegarder : avant de quitter le programme saisi, il faut l'enregistrer, et ça peut se faire en cliquant sur cette icône.



Transférer ou téléverser vers la carte : ce bouton permet de compiler votre code et de le transférer vers la carte Arduino.



Moniteur série : pour ouvrir la fenêtre du moniteur série il suffit d'appuyer sur ce bouton.

1.2.2.3. Description de la barre des menus :

Le logiciel Arduino est équipé d'une barre dit des menus qui contient un ensemble de menus (d'où le nom «Barre des menus») permettant de faire la gestion du programme.



Figure 9: la barre des menus

Cette barre contient les icônes suivantes:

- **File (Fichier):** ce menu contient les différentes options de création, d'ouverture, de sauvegarde, d'impression d'un programme, ou l'ouverture d'un exemple parmi les exemples qui accompagnent le logiciel Arduino.
- **Edite(Editer):** ce menu contient les options de copier/coller, sélection, et les options de recherche.
- **Sketch (Programme ou séquence):** ce menu contient les différentes fonctions de la barre des boutons, ainsi que les options d'ajout de bibliothèques ou de fichiers.
- **Tools (Outils):** c'est dans ce menu qu'on sélectionne le type de carte à programmer, et le port série utilise ainsi que la fonction de chargement du boot loader dans l'ATmega.
- **Help (Aide):** ce menu est fait pour donner de l'aide concernant les différents problèmes rencontrés au niveau du logiciel Arduino.

1.2.2.4. Les étapes de téléversement d'un programme vers la carte Arduino :

Une fois le programme est saisi (première phase), une deuxième phase consiste à vérifier s'il y a des erreurs dans le programme, et pour la faire il suffit de cliquer le bouton de **vérification** se trouvant à l'extrémité gauche de la barre des boutons et attendre un petit moment.

Après cette vérification s'il y a des erreurs, un message en orange s'affiche au dessous de l'onglet d'édition des programmes indiquant qu'elle est l'erreur et la ligne où se trouve. Une fois la correction des erreurs est terminée, on passe à la phase de la sélection du type de la carte à programmer et le port série sur lequel est branchée cette carte. Ce petit réglage peut se faire à l'aide du menu **Tools(Outils)**. Une fois terminé il reste qu'à cliquer sur le bouton de **Téléversement** qui se trouve dans la barre des boutons juste avant le bouton du moniteur série.

Si tout marche bien les deux LEDs Rx et Tx de la carte s'allument, sinon un message d'erreur s'affiche dans la console des messages de compilation.

A ce niveau la carte est programmée et prête pour l'exécution du programme.

1.2.3. La carte Eye-Fi

1.2.3.1. Eye-Fi Connect X2 4Go :

Le fabricant Eye-Fi a conçu, depuis quelques années déjà, des cartes mémoires wifi pour appareils photos numériques. C'est le modèle **Eye-Fi Connect X2 4Go**, au format SD Card, que nous allons passer en revue.



Figure 10:Eye-Fi Connect X2 4Go

Il s'agit du modèle Eye-Fi Connect X2 du constructeur qui propose deux autres cartes wifi la *Pro X2* et la *Mobile X2*, de capacité 8Go.

Un des avantages d'une telle carte, c'est qu'elle permet de télécharger automatiquement, au moyen du wifi, les photos qu'elle contient vers un ordinateur ou un espace de stockage en ligne .

De plus, ce modèle de carte est capable de mémoriser des informations de localisation géographique pour chaque prise de vue, afin de permettre une localisation GPS de la photo.

Le fabricant annonce une compatibilité avec un très grand nombre d'appareils photos numériques du marché, et revendique une installation et une utilisation d'une très grande simplicité. De surcroît, des applications pour appareils mobiles sous iOS et Android sont disponibles, pour favoriser à la fois la diffusion et la visualisation des photos réalisées.

Pour l'installation il suffit d'insérer directement le lecteur de carte USB dans un ordinateur de type PC ou Mac. Ce sont ces indications qui sont données dans le manuel d'installation fourni qu'elle faut suivre. La configuration requise est un ordinateur fonctionnant sous Windows XP (SP3)/Vista/7/8, sous Mac OSX (Intel) 10,5 ou sous 10,6 (Snow Leopard). Environ 100 Mo sont nécessaires sur le disque dur de l'ordinateur pour l'installation du logiciel.

1.2.3.2. Caractéristiques techniques de l'Eye-Fi Connect X2 :

La carte est au format SD Card, d'une capacité totale de 4 Go, et de classe SDHC 6, c'est-à-dire un indice de rapidité minimale d'écriture de 6Mo/s.

La carte supporte les réseaux Wifi en 802.11 b/g/n dans la bande des 2,4Ghz exclusivement. Le constructeur indique clairement sur son site que la bande Wifi des 5Ghz n'est pas supportée.

Les cryptages Wifi supportés sont

CHAPITRE II

- le WEP statique en 64 et 128 bits.
- le WPA-PSK.
- le WPA2-PSK.

Il est possible d'enregistrer la carte sur plusieurs réseaux wifi au moyen du logiciel (*Eye-Fi Center*) fourni sur cette dernière.

La portée du Wifi de la carte est annoncée pour plus de 90m en extérieur et de 45 mètres en intérieur. Afin de limiter la consommation énergétique de l'appareil photo, le constructeur annonce avoir mis en place un dispositif optimisant les ressources nécessaires pour la carte, sans plus de détails. Il faut bien avoir conscience que c'est la batterie de l'appareil photo numérique qui va fournir l'énergie nécessaire à la transmission par Wifi.

Outre le transfert de photos au format JPG ou RAW, la carte est également capable de transmettre des vidéos, que ce soit vers un ordinateur ou vers des sites de partage en ligne. Pour les vidéos, il ne s'agit pas de transferts « en direct » de la vidéo que l'on réalise (c'est-à-dire en cours d'écriture), mais plutôt du fichier vidéo réalisé (c'est-à-dire fini d'écrire).

1.2.3.3. Description du logiciel :

Après avoir installé le logiciel Eye-Fi, la carte SD est alors détectée par le logiciel comme étant une carte Eye-Fi :

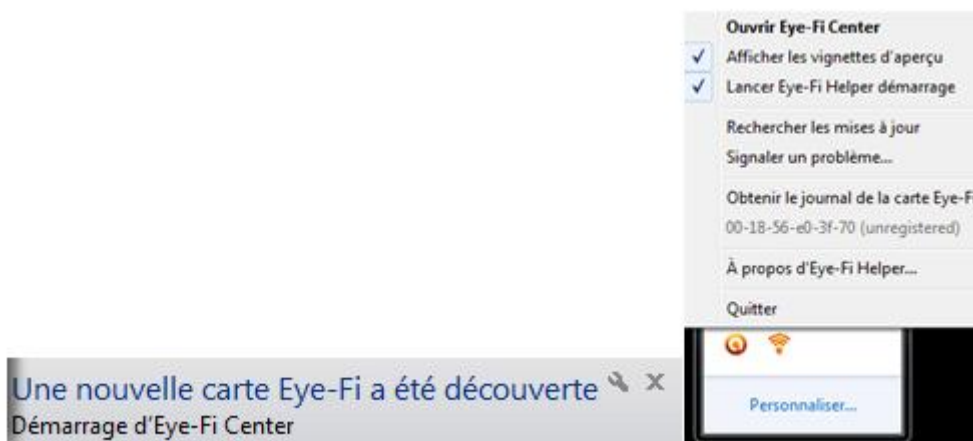


Figure 11:PC annonce qu'une nouvelle carte Eye-fi a été découverte

On peut enfin lancer l'Eye-Fi center, qui s'ouvre sur un message d'astuce d'utilisation, logiciel est alors fonctionnel.

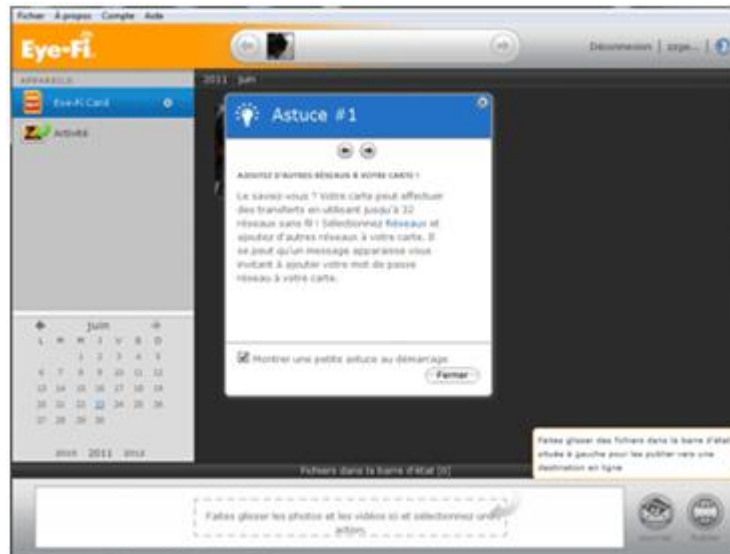


Figure 12: Application Eye-fi

1.2.3.4. Paramétrage de la carte Eye-Fi :

1.2.3.4.1. Les Wifi réseaux privés et réseaux publics

Pour pouvoir paramétrer la carte Eye-Fi, celle-ci doit être présente dans n'importe quel lecteur de carte SD. Celui fourni par le constructeur n'étant pas obligatoire.

Il s'agit tout d'abord de définir les réseaux wifi auxquels la carte peut se connecter. Plusieurs protocoles de cryptage sont supportés et il est possible de faire détecter tous les réseaux environnants ou de saisir manuellement un réseau qui serait hors de portée :

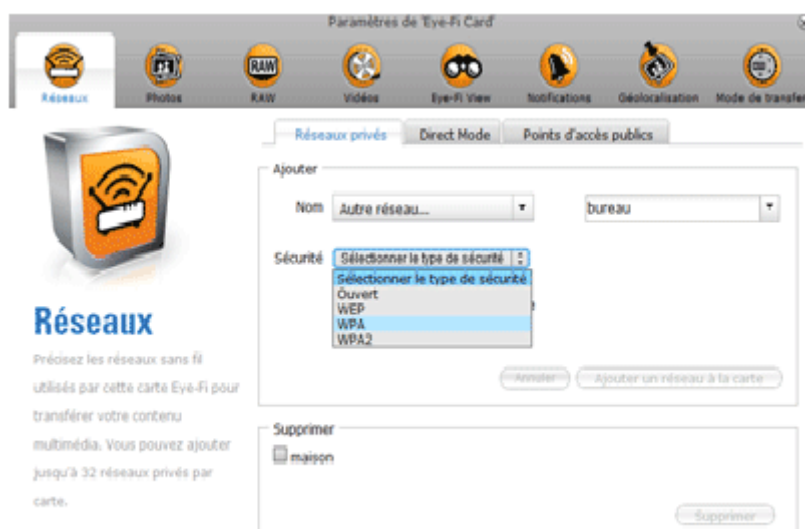


Figure 13: Paramétrage wifi des réseaux privés : Ouvert, Wep, WPA

CHAPITRE II

Il est ainsi possible de définir jusqu'à 32 réseaux privés auxquels la carte sera autorisée à se connecter.

Mais lorsqu'on est en extérieur, il est également possible de se connecter à des réseaux ouverts publics. Pour cela, une option est disponible pour indiquer si la carte est autorisée à transférer les photos via ce type de réseau.



Figure 14: autorisation de transfert via un réseau

1.2.3.4.2. Le Wifi Direct-Mode

Enfin un troisième mode de transfert Wifi est disponible : il s'agit du *Direct Mode*. Dans ce mode, la carte Eye-Fi crée un réseau Ad-hoc protégé en WPA2, à la norme Wifi 802.11n sur lequel un appareil sous iOS ou Android (du type Smartphone, iPod, tablettes, etc.) pourra se connecter.



Figure 15: Paramétrage du Wifi en Direct Mode

CHAPITRE II

Les photos y seront alors directement transférées.

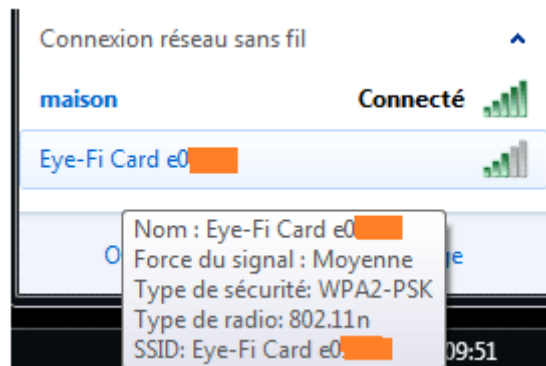


Figure 16: Réseau Ad-hoc créé par la carte Eye-Fi

Eye-Fi propose sur son site les applications appropriées pour chacun de ces environnements, permettant notamment de les visualiser sur son appareil mobile.

S'il s'agit d'un appareil accédant à Internet avec un débit suffisant (par exemple 3G en HSDPA), il devient alors possible, même en pleine nature, d'envoyer vers les sites de partage, ses prises de vues, ou tout simplement de commencer à regarder les photos réalisées sur un écran plus grand que celui de l'appareil ayant servi à la prise de la photographie. Il devient alors possible, en photographie paysagère par exemple, d'envisager un travail de terrain un peu plus approfondi en décidant, le cas échéant, de réaliser un nouvelle série de vues.

1.2.3.5. Sélection et gestion des formats transférés :

Les appareils photos, suivant les modèles, prennent les photos au format JPEG et RAW. Il est possible de définir pour chacun de ces formats les options de transferts et de stockage distant, et cela de manière indépendante l'une de l'autre.

Ainsi, les deux onglets *Photos* et *RAW* proposent chacun deux écrans de paramétrage analogues où l'on va pouvoir définir l'emplacement de stockage sur l'ordinateur qui réceptionnera les photos et la manière dont celles-ci seront triées : un sous-dossier par date, le format de date étant personnalisable, ou tout mettre dans un seul et unique dossier.



Figure 17: Organisation du dépôt des photos sur l'ordinateur

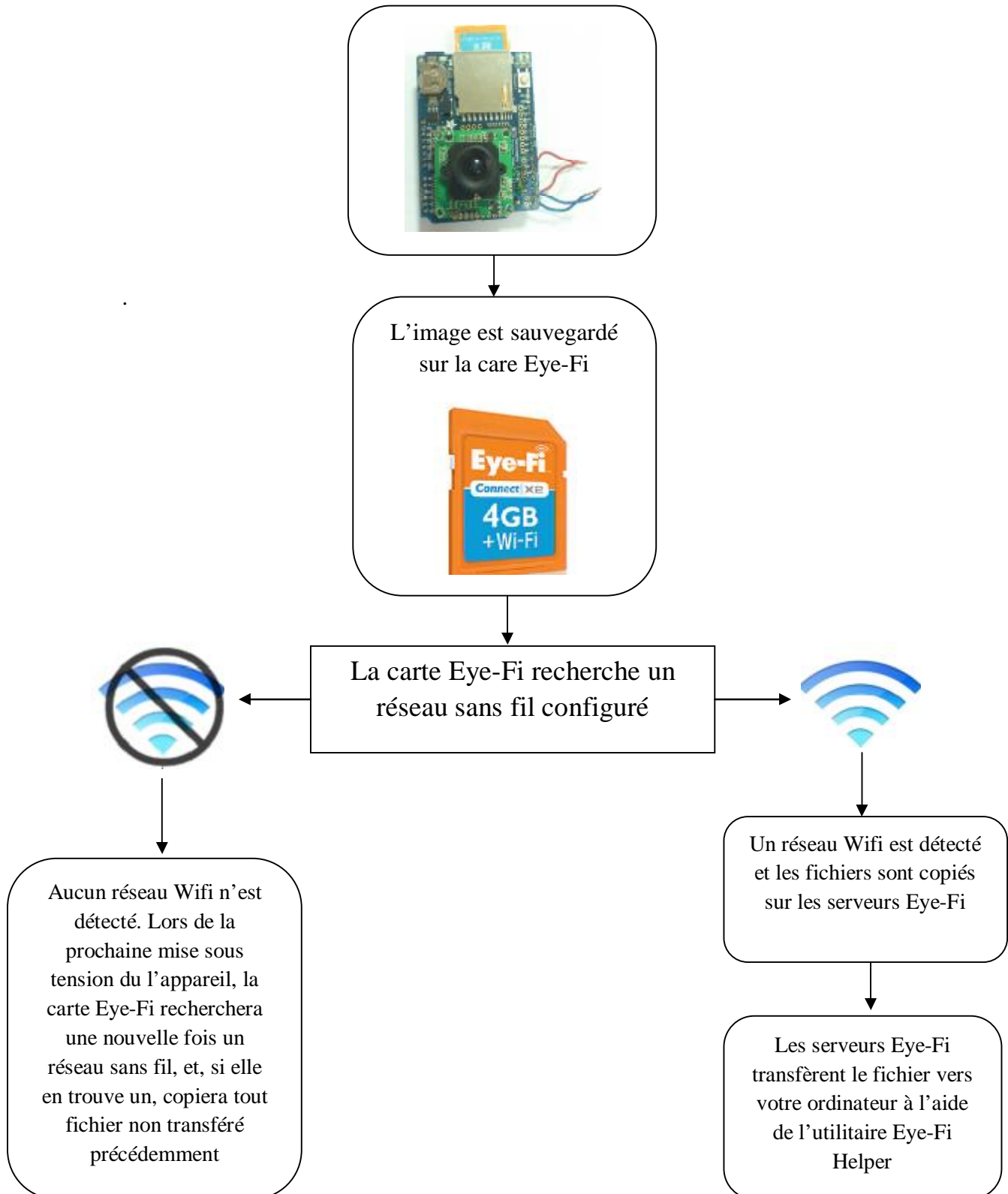
La publication sur Internet peut se faire par le biais de plates-formes populaires telles que Facebook, MobileMe ou encore Picasa



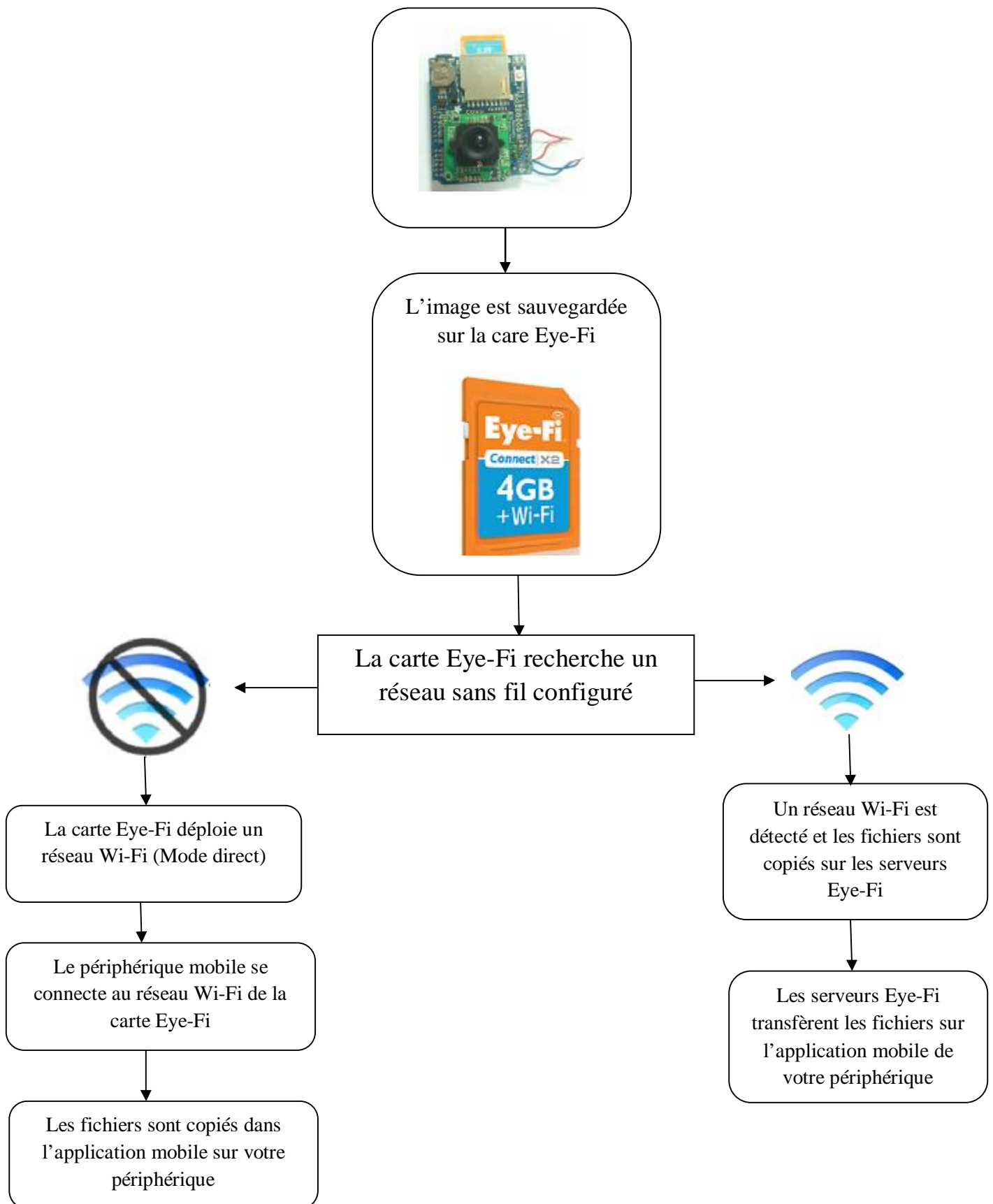
Figure 18: Publication sur internet

1.2.3.6. Principe de fonctionnement d'Eye-Fi :

- Envoyer des fichiers vers un ordinateur



- **Envoyer des fichiers vers un périphérique mobile :**



1.2.4. Shield carte SD :

1.2.4.1. Description du shield carte SD :

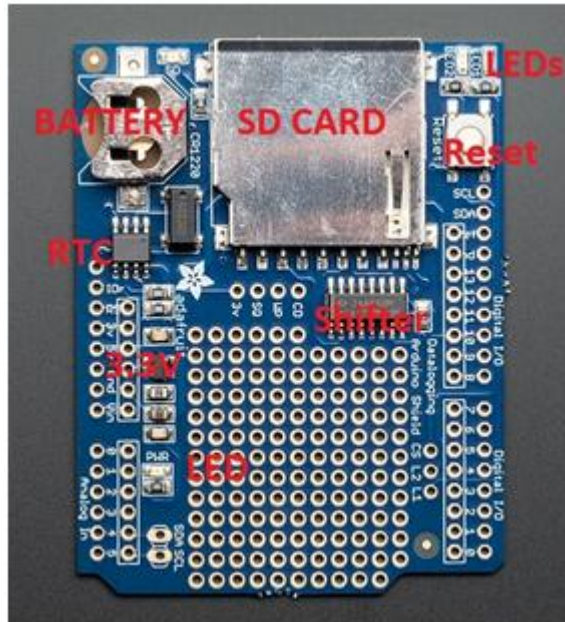


Figure 19:vue de face d'un shield SD carte

1.2.4.2. Caractéristiques:

- interface de carte SD fonctionne avec FAT16 ou FAT32 cartes formatées. Niveau de 3.3v circuit de décalage évite d'endommager la carte SD
- Horloge temps réel (RTC) maintient le temps, même lorsque l'Arduino est débranché. La batterie de secours peut durer des années
- Bibliothèques incluses et un exemple de code à la fois SD et RTC
- Zone pour les connecteurs à souder, des circuits ou des capteurs de prototypage.
- LEDs de signalisation configurables
- Régulateur de 3,3 V à bord et à la fois une tension de référence fiable

1.2.4.3. Les principaux composants du shield :

- Il a une horloge temps réel (RTC) qui a une puce, de cristal et une batterie de secours
- Un régulateur carte de 3,3 V pour faire fonctionner les pièces de 3V
- PWR vert (alimentation) bon LED Haut Moyen
- Un grand détenteur de la carte SD peut s'adapter à n'importe quel stockage SD / MMC de 32Mo a 32Go

CHAPITRE II

- Un bouton de réinitialisation permet de réinitialiser l'ensemble Arduino

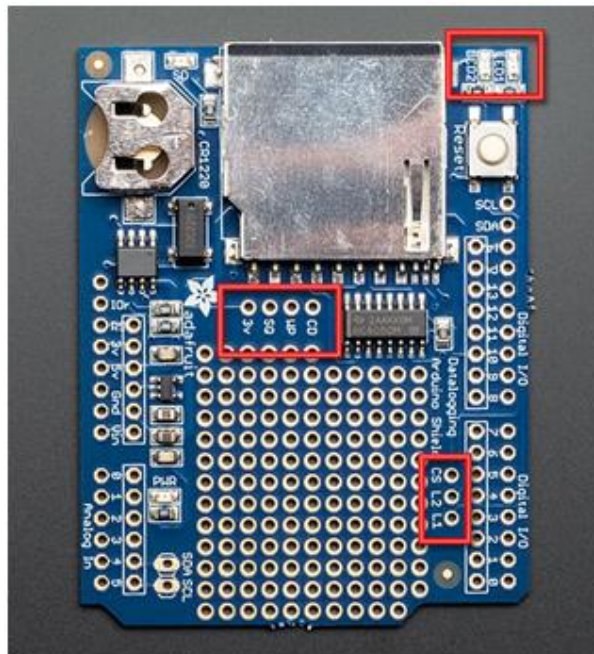


Figure 20:les composants du shield

- **3V** - c'est le 3V sur le régulateur. C'est une référence de 3,3 V de bonne pour alimenter des capteurs de puissance. Jusqu'à 50 mA est disponible
- **SQ** - c'est la sortie Squarewave option du CCF. On doit envoyer la commande à activer cette option, mais c'est un moyen d'obtenir éventuellement un créneau de précision. On l'utilise principalement pour les tests
- **WP** -c'est la protection en écriture tampon sur la carte SD, on peut l'utiliser pour détecter si l'onglet de protection en écriture est sur la carte en cochant cette broche
- **CD**- c'est la carte de détecter tampon sur la carte SD. Lorsque cela est lié à la terre, une carte SD est insérée.
- **CS**- c'est le Chip Select broches pour la carte SD. Si on doit couper la trace à la broche 10, car il est en conflit, ce tapis peut être soudé à une broche numérique et le logiciel re-téléchargés.
- **L2** et **L1** - ce sont des utilisateurs-LED en option. Connectez-vous à n'importe quelle broche numérique, tirez élevée pour faire allumer la LED correspondante. Les LED ont déjà 470 ohms résistances en série.

1.2.5. Camera série TTL :

1.2.5.1. Description de la camera série TTL°:

Le module série TTL de l'appareil avec sortie vidéo NTSC. Les modules ont quelques fonctionnalités intégrées, telles que la capacité de changer la luminosité, saturation, teinte des images, des auto-ajustements du contraste et de luminosité automatique et détection de mouvement.



Figure 21:camera série TTL

Le module a été initialement conçu à des fins de surveillance. Les commandes du port série peuvent demander que le module de geler la vidéo et ensuite de télécharger une image en couleur JPEG. Ainsi, Lorsqu'un mouvement est détecté, il prend une photo et l'enregistrer sur un disque pour une analyse ultérieure.

La taille d'image maximale qu'il peut prendre est de 640x480 pixels. Et il est sensible à la lumière infrarouge, ce qui modifie quelque peu le rendu des couleurs. La raison de tout cela, c'est qu'il est destiné à la surveillance.

- **Taille du module:** 32mm x 32mm
- **Capteur d'image:** CMOS 1/4 de pouce
- **Pixels CMOS:** 0,3 M
- **La taille du pixel:** 5.6um * 5.6um
- **Format de sortie:** JPEG standard / M-JPEG
- **Balance des blancs:** automatique
- **Exposition:** automatique

CHAPITRE II

- **Gain:** Automatique
- **Shutter:** volet roulant électronique
- **SNR:** 45DB
- **Dynamic Range:** 60DB
- **Gain analogique Max:** 16dB
- **vitesse de la monture:** 640 * 480 30fps
- **Mode de balayage:** balayage progressif
- **Angle de vision:** 60 degrés
- **Surveillance à distance:** 10 mètres, maximum de 15mètres (réglable)
- **Taille de l'image:** VGA (640 * 480), QVGA (320 * 240), QQVGA (160 * 120)
- **vitesse de transmission:** par défaut 38400
- **Consommation de courant:** 75mA
- **Tension de fonctionnement:** DC +5 V
- **Communication:** 3,3 V TTL (Trois fils TX, RX, GND)

1.2.5.2. Câblage de l'appareil photo

Le module est livré sans connecteur de sort, on aura besoin donc de souder des fils dans les plots de connexion. On a utilisé le rouge pour la broche +5 V, noir pour la broche GND blanc pour les broches RX et vert pour la broche TX

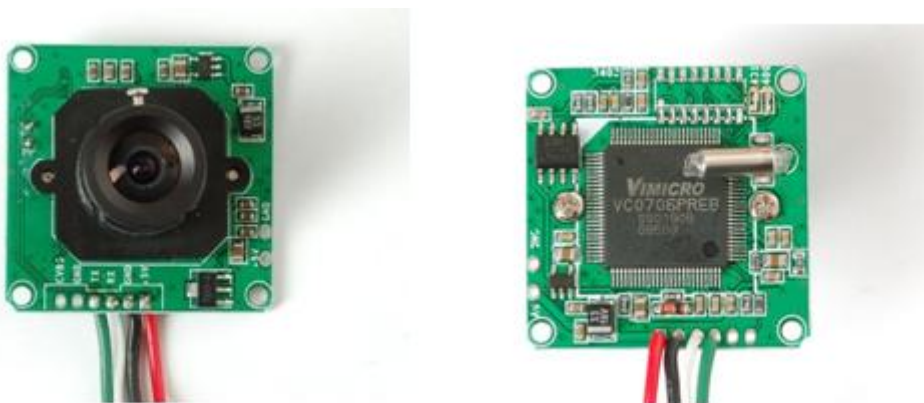


Figure 22: camera série TTL

Si on souhaite obtenir la vidéo NTSC à sur un moniteur, on soude un fil noir à la seconde broche GND, et un fil jaune à la broche CVBS.



Figure 23: camera série TTL avec sortie vidéos

1.2.5.3. Utilisation de l'appareil

Pour l'utilisation de la camera, on a raccordé l'appareil à notre microcontrôleur dans notre cas un Arduino. On a utilisé quatre broches numériques ainsi les broches série RX et TX de la carte Arduino. Pour enregistrer des images, on a utilisé une carte mémoire de type SDHC Eye-Fie connect X2, placé sur un module de carte SD arduino.

1.2.5.4. Détection de mouvement :

Une chose intéressante et que la caméra est construite avec la capacité de détection de mouvement. Elle va chercher le mouvement dans le flux vidéo et d'alerter le microcontrôleur (par l'envoi d'un paquet de données série) lorsqu'un mouvement est détecté

Une caméra à détection de mouvement est un capteur de mouvement basé sur l'analyse d'images en continu. C'est une caméra disposant d'un système de détection automatique des gestes, ce qui active la prise de vue dès que le capteur détecte un mouvement.

1.3. Tests et résultats obtenus

1.3.1. Tests de captures

1.3.1.1. Résolution



Figure 24: Image 640X480



Figure 24: Image 320X240



Figure 26: Image 160X120

Comme on peut le voir sur les trois images, les différentes résolutions donnent des images d'assez bonne qualité ce qui nous offre un espace de sauvegarde assez important sur la carte SD et une vitesse de transmission assez bonne en optant pour la moyenne résolution.

1.3.1.2. Vitesse de déplacement de la cible



Figure 26: Personne en déplacement



Figure 25: Objet en chute libre

La figure (27) illustre la capture d'une personne en mouvement (vitesse moyenne) et la figure (28) un objet en chute libre. Comme vous pouvez le constater, la capture est bonne même pour des objets en mouvement à grande vitesse.

1.3.2. Tests de transmission

1.3.2.1. Transmission vers un compte (type facebook)



Figure 27: images de notre teste après transmission vers Facebook

On reconnaît bien le site Facebook vers lequel les images captées ont été transmises. Ce système peut transmettre des images vers d'autres supports tels que les micros ordinateurs, les Smartphones, les bases de données etc....

1.3.2.2. Transmission vers un micro ordinateur

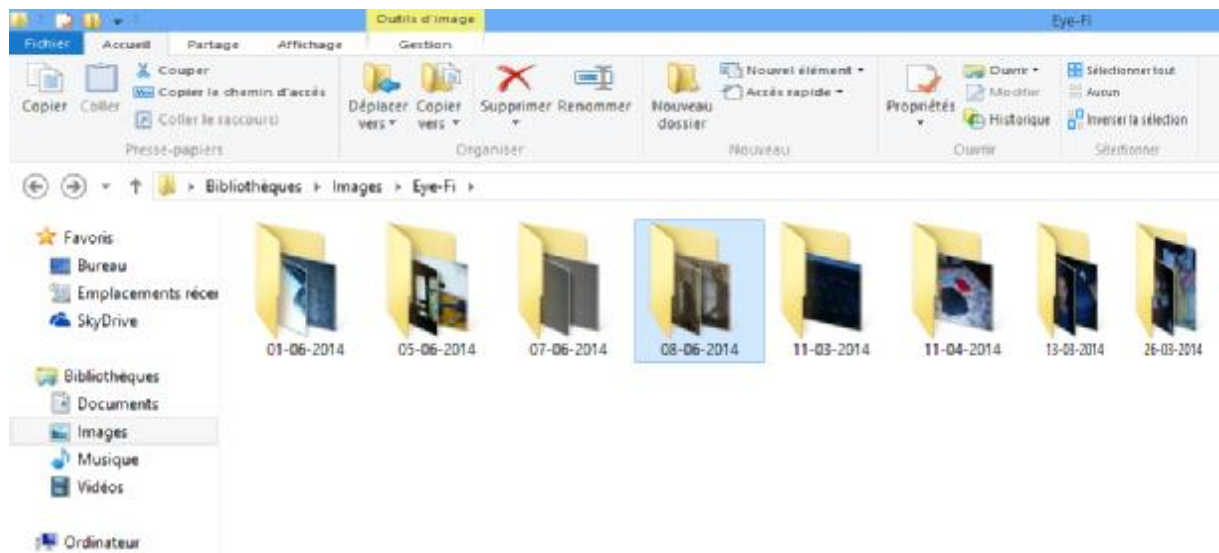


Figure 30: dossiers de sauvegardes

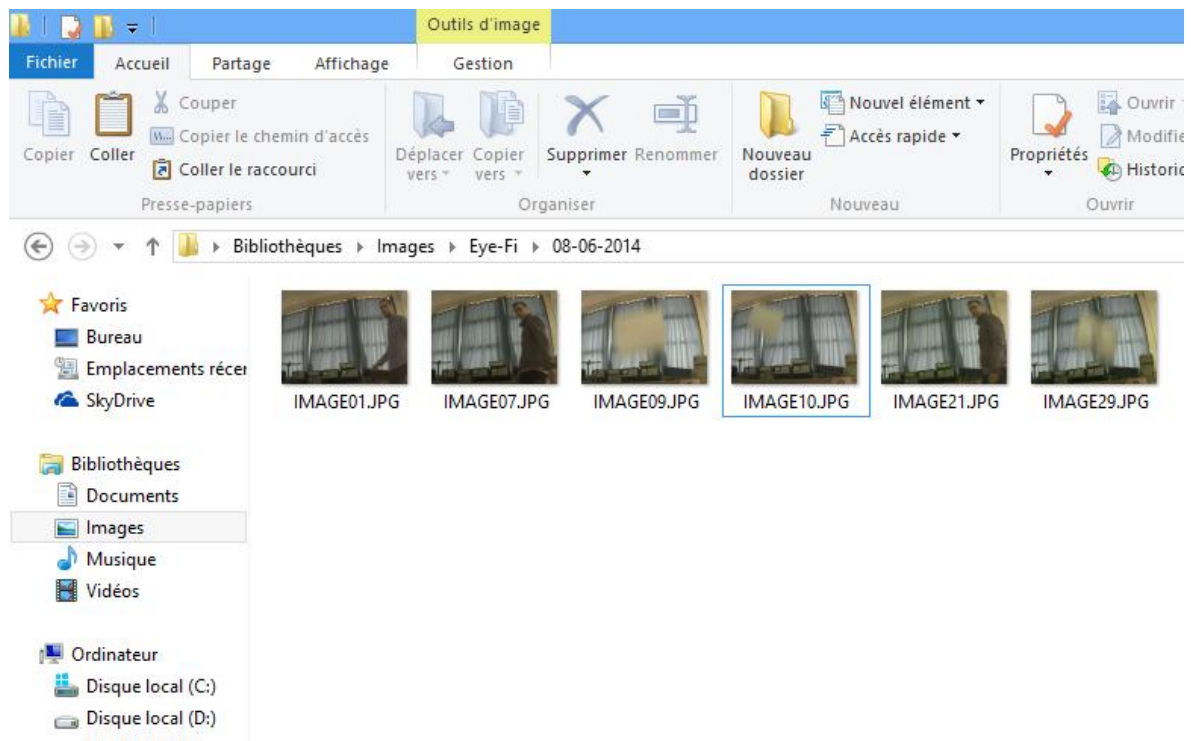


Figure 28: contenus des sauvegardes

1.3.2.3. Transmission vers un smart phone



Figure 29: application Eye Fi sous un Smartphone



Figure 30: images transmises vers un Smartphone

1.3.3. Procédure de transmission :

Après la détection d'un mouvement par notre système, une image est prise et sauvegardée sur la carte Eye Fi pour sa transmission.

La transmission s'effectue selon deux procédures :

- Si une connexion internet wifi est disponible, notre système utilise le mode publique ou privé pour envoyer l'image prise vers Facebook ou smart phone via le serveur Eye-Fi.
- Si une connexion internet wifi n'est pas disponible, le système utilise le mode direct pour transmettre l'image prise vers un micro ordinateur.

Conclusion

On assiste actuellement au développement des systèmes de communications à vitesse grand V. l'échange d'informations et leurs normalisations sont au centre des préoccupations des concepteurs. Dans cette course aux nouvelles techniques de communications, les communications sans fils occupent une place importante.

Le travail demandé consiste à proposer un système de vidéosurveillance sans fils, performant et lowcost à base de la carte Arduino.

Les résultats obtenus sont très satisfaisants, en capture et en transmission, ce qui rend notre système prêt à l'emploi. Il peut être commercialisé à ce stade. Cependant, notre système peut être amélioré, ce qui présente de bonnes perspectives pour le travail.

Les améliorations qu'on souhaite apporter à notre système peuvent être résumées comme suit :

- Création d'un serveur de base de données propre à notre application et rajouter une application de traitement d'images pour la reconnaissance de visages pour donner plus de mobilité au personnel travaillant dans la zone sous vidéosurveillance (authentification par reconnaissance faciale) et ne pas transmettre inutilement des images.
- Elaboration d'une application pour la sauvegarde et la transmission des données propre à nous.
- Prévoir un système d'alarme pour alerter les personnes concernées.

En fin, le travail demandé nous a permis d'approfondir nos connaissances dans le domaine de programmation et de prendre en main la carte ARDUINO.

Pour conclure nous espérons que notre travail puisse servir de support pour les promotions à venir et inciter nos camarades étudiants à s'intéresser davantage aux systèmes embarqués.

Bibliographie

Livres :

Nom : MONK Simon

Titre : 30 ARDUINO projects for the evilgenius

Edition :Mc Graw Hi (2010).

Nom :BERTRAND Cottenceau,

Titre : Carte ARDUINO UNO microcontrôleur ATM328

Edition :ISTIA(2010).

Nom : EVANSBrian

Titre :Beginning Arduino programming

Edition: APRESS (2011).

Nom :BENZI Massimo

Titre : GettingStartedWith Arduino

Edition : ORELLY (2011).

Sites internet :

<http://www.eyefi.com/>

<http://www.arduino.cc/>

<http://www.adafruit.com/>

<http://www.wikipedia.org/>

<http://www.datasheetcatalog.com/>

```

#if ARDUINO >= 100
  #include <SoftwareSerial.h>
#else
  #include <NewSoftSerial.h>
#endif
#include <Adafruit_VC0706.h> // librairie de la camera ttl
#include <SPI.h>
#include <SD.h> // librairie de la carte sd
#include <RTCLib.h> // librairie de temps reel
#include <Wire.h> // nécessaire pour RTC

#define RED_LED 6
#define GREEN_LED 5

// contient les niveaux de luminosité dans le temps
PROGMEM byte sleepTab[] = {
  0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
  1, 1, 2, 3, 4, 5, 6, 8, 10, 13,
  15, 19, 22, 26, 31, 36, 41, 47, 54, 61,
  68, 76, 84, 92, 101, 110, 120, 129, 139, 148,
  158, 167, 177, 186, 194, 203, 211, 218, 225, 232,
  237, 242, 246, 250, 252, 254, 255 };

#define chipSel 10

RTC_DS1307 clock;
Adafruit_VC0706 cam = Adafruit_VC0706(&Serial);
char directory[] = "DCIM/CANON999",
layout
filename[] = "DCIM/CANON999/IMG_0000.JPG";
byte sleepPos;
int imgNum = 0;
const int minFileSize = 20 * 1024;
size

// -----
---

void setup() {

  pinMode(RED_LED, OUTPUT);
  pinMode(GREEN_LED, OUTPUT);
  digitalWrite(RED_LED, LOW);
  digitalWrite(GREEN_LED, HIGH); // "bonjour!" indicator

hardware

  pinMode(2, INPUT); pinMode(3, INPUT);

  Wire.begin();
  clock.begin();
RTCLib.
  SdFile::dateTimeCallback(dateTime);
  if(!clock.isrunning()) error(75);
  if(!SD.begin(chipSel)) error(250);
flash
  if(!cam.begin()) error(1000);

```

```

SD.enableCRC(true);

    if(!SD.exists(directory) && !SD.mkdir(directory)) error(1);

delay(1000);
cam.setImageSize(VC0706_320x240);

    TCCR1A = _BV(WGM11); // Mode 14 (fast PWM), 64:1 prescale, OC1A off
    TCCR1B = _BV(WGM13) | _BV(WGM12) | _BV(CS11) | _BV(CS10);
    ICR1   = 8333;
    sei();
}

void loop() {

    sleepPos = sizeof(sleepTab);
    TIMSK1  |= _BV(TOIE1);
    nextFilename();
    cam.resumeVideo();
    cam.setMotionDetect(true);
    while(!cam.motionDetected());

    // Motion detected!
    cam.setMotionDetect(false);
    TIMSK1 &= ~_BV(TOIE1);
    digitalWrite(GREEN_LED, HIGH);

    delay(500);
    if(!cam.takePicture()) {
        // Failed to take picture. Show RED (+GREEN above) for 5 sec:
        digitalWrite(RED_LED, HIGH);
        delay(5000);
        digitalWrite(RED_LED, LOW);
        return; // Resume motion detection
    }

    File imgFile = SD.open(filename, FILE_WRITE);
    if(imgFile == NULL) {
        // Couldn't open file. Show RED (no GREEN) for 5 sec:
        digitalWrite(GREEN_LED, LOW);
        digitalWrite(RED_LED, HIGH);
        delay(5000);
        digitalWrite(RED_LED, LOW);
        return; // Resume motion detection
    }

    uint16_t jpegLen = cam.frameLength();
    uint16_t bytesRemaining;
    uint8_t b, *ptr;

    // Transfer data from camera to SD file:
    for(bytesRemaining = jpegLen; bytesRemaining ; bytesRemaining -= b) {
        b = min(32, bytesRemaining); // Max of 32 bytes at a time
        ptr = cam.readPicture(b); // From camera
        imgFile.write(ptr, b); // To SD card
        digitalWrite(GREEN_LED, (bytesRemaining & 256) ? HIGH : LOW);
    }
}

```

```

}

// Pad file to minimum Eye-Fi file size if required:
if(jpegLen < minFileSize) {
    for(bytesRemaining = minFileSize - jpegLen; bytesRemaining ;
        bytesRemaining -= b) {
        b = min(32, bytesRemaining);
        imgFile.write(ptr, b);
        digitalWrite(GREEN_LED, (bytesRemaining & 256) ? HIGH : LOW);
    }
}

imgFile.close();
}

// Handler for unrecoverable errors (e.g. during hardware init).
// Flashes red LED at given speed. Does not return!
void error(int time) {
    digitalWrite(GREEN_LED, LOW);
    for(;;) {
        digitalWrite(RED_LED, HIGH);
        delay(time);
        digitalWrite(RED_LED, LOW);
        delay(time);
    }
}

void nextFilename(void) {
    for(;;) {
        filename[18] = '0' + imgNum / 1000;
        filename[19] = '0' + (imgNum / 100) % 10;
        filename[20] = '0' + (imgNum / 10) % 10;
        filename[21] = '0' + imgNum % 10;
        if(!SD.exists(filename)) return; // Name available!
        imgNum++; // Keep looking
    }
}

void dateTime(uint16_t* date, uint16_t* time) {
    DateTime now = clock.now();
    *date = FAT_DATE(now.year(), now.month(), now.day());
    *time = FAT_TIME(now.hour(), now.minute(), now.second());
}

// Timer1 interrupt handler for sleep throb
ISR(TIMER1_OVF_vect, ISR_NOBLOCK) {
    // Sine table contains only first half...reflect for second half...
    analogWrite(GREEN_LED, pgm_read_byte(&sleepTab[
        (sleepPos >= sizeof(sleepTab)) ?
        ((sizeof(sleepTab) - 1) * 2 - sleepPos) : sleepPos]));
    if(++sleepPos >= ((sizeof(sleepTab) - 1) * 2)) sleepPos = 0; TIFR1 |=
TOV1;
}

```