

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE, UNIVERSITE MOULOU MAMMERI, TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE

DEPARTEMENT D'ELECTRONIQUE

Mémoire de fin d'études

En vue de l'obtention du diplôme de Master en Electronique

Option : Réseaux et Télécommunication

Thème :

*Conception et réalisation d'un système de
télésurveillance médicale à base d'une carte*

ARDUINO

"Transmission bluetooth"

Proposé par :

Pr .M.LAGHROUCHE

Présenté par :

NACHEF MHENNA

Année universitaire 2012/2013

Remerciement

Je remercie ALLAH le tout puissant de m'avoir donné le courage et la volonté de mener à terme ce présent travail.

Je remercie énormément et infiniment Monsieur M.LAGHROUCHE d'avoir accepté de m'encadrer et je lui suis très reconnaissant pour ces précieuses aides pendant les moments difficiles de ma recherche.

Je tiens aussi à remercier Mr .H.Hamiche et Mr.K.Benamane pour leurs conseils et leurs dévouement afin d'achever à bien ce mémoire.

Je tiens à remercier les membres de jury qui ont acceptés de juger ce modeste travail.

Dédicace

Je remercie Dieu le tout puissant d'avoir exaucé mon vœu de réaliser ce projet de fin d'études.

Je voudrais dédier le présent travail tout spécialement à mes chers parents qui m'ont élevé et soutenu tout au long de ma vie.

Je dédie également ce projet à mes chères sœurs Samira, Nabila, Aicha, Assia et ses petits enfants Abd eraoufe et Maya, ainsi que ma grande mère et toute la famille.

Je tiens énormément à remercier mes chers amis et camarades de ma promotion surtout ceux avec lesquels j'ai travaillé au Laboratoire projet : Rabah, Hichem, Sofiane, Djamel amazighe, Rafik, Kouci et Mustafa.

Sans oublier mes chers amis et copains, Akli, Zohere, Abd elghani, bilaide, oussamah, yacine, ziz, elakhe Djamel, Toufik, lounas, Djamel ahnouche et djamel boudjnah.

Je voudrais aussi dédier cette thèse au responsable du laboratoire projet Mme Slimani.

Enfin, je voudrais dédier cette thèse à toute personne ayant participé de loin ou de près à la réalisation de ce travail.

Sommaire

Introduction générale

Chapitre I : Les principes de base du signal ECG

I-Préambule	1
II-1-Anatomie et fonctionnement du cœur	1
II-2-Le cycle cardiaque	3
II-2-1-Le potentiel d'action	4
II-2-2-Activation cardiaque physiologique et conduction intracardiaque	5
III-L'électrocardiogramme	8
III-1-1Historique	8
III-2-Enregistrement de l'électrocardiogramme.....	8
III-2-1-L'ECG de Holter	10
III-3-Les électrodes	11
III-3-1-Emplacement des électrodes.....	11
III-4-Tracé de l'activité électrique du cœur	12
III-4-1-Intervalles temporels	13
IV-Conclusion	15

Chapitre II : Acquisition et transmission du signal ECG

I-Introduction	16
II-Description du système.....	16
II-1-L'amplificateur d'instrumentation	17
II-2-Circuit du pied droit (Righ-Leg-Drive)	20
II-3-Circuit de fixation de niveau	21
II-4-Le circuit de filtrage	21
III-Description de la carte ARDUINO UNO	22

III-2-Alimentation de la carte Arduino	23
III-2-1-Protection du port USB contre la surcharge en intensité.....	25
III-3-Gestion des mémoires dans la carte ARDUINO	25
III-4-Les entrées/ sorties numériques	25
III-5-Les entrées analogiques.....	26
III-6- ARDUINO et la communication avec l'extérieur	27
IV- Le microcontrôleur ATmega328	29
IV-1 Les principales caractéristiques de l'ATmega328	29
IV-2-Le capteur de température /humidité SHT75	30
V-1- Les caractéristiques du SHT75	31
V-2- La structure interne du capteur	31
V-3- Conversion de la sortie numérique en valeurs physiques	32
V-4- Description du SHT75	34
V-5- La communication avec le SHT75.....	36
V-5-1 La mesure de l'humidité et de la température	38
VI- Le module de transmission	39
VI-1-Historique.....	39
VI-2 Le module bluetooth de Sparkfun	39
VI-2-1 Les caractéristiques du module BlueSMiRF	40
VI-2-2 Brochage du module BlueSMiRF	40

Chapitre III Programmation et conception logicielles

I- Introduction	45
II- l'environnement de programmation Arduino	45
II-1- Description du logiciel Arduino	46
II-1-1 Description de la barre des boutons	48
II-1-2-Description de la barre des menus	49
III- Les étapes de téléversement d'un programme vers la carte Arduino.....	50
IV- Le logiciel BlueSoleil	51

IV-1 Description de BlueSoleil	52
V- Le logiciel de programmation graphique LabVIEW	53
V-1- Les instruments virtuels	54
V-2 Les types de données de base utilisés dans LabVIEW	56
V-3- Description de notre interface réalisée	57
V-3-1- Configuration du port.....	58
V-3-2- Configuration de la taille du tampon d'entrée/sortie	59
VI Résultat	63
Conclusion	
Bibliographie	
Annexe	

Introduction générale

INTRODUCTION

Beaucoup de gents souffrent des maladies du cœur appelées maladies cardiaques qui se traduisent souvent par une anomalie dans le comportement du cœur et qui mettent le patient sous le danger d'arrêt cardiaque. Un diagnostic alors au niveau du cœur s'impose, pour déduire la source et le type de cette anomalie, et pour se faire, une solution simple consiste à lire le signal d'ELECTROCARDIOGRAMME qui est un signal traduisant le rythme cardiaque par un phénomène électrique qui peut être recueilli sur la peau de l'individu.

En raison des risques de ce type de maladies, un contrôle continu est obligatoire pour suivre l'état du malade. A ce stade on fait appel à un domaine un peut particulier qui est la télésurveillance médicale.

La télésurveillance médicale est une branche de la télémédecine (Télécommunication + médecine) qui a pour but de surveiller à distance les paramètres d'un patient, ce ci est très important pour fournir l'assistance spécialisée pour des personnes qui vivent dans des endroits éloignés et les personnes qui ont besoin d'une surveillance continue dans les environnements non-cliniques (c.-à-d. à la maison, travail) , étant donné que c'est l'objectif de notre étude, cette technique permet le suivi en temps réel de l'état du patient, grâce à des systèmes électroniques qui intègrent au même temps l'acquisition, le traitement, l'enregistrement, et la transmission du signal d'ECG.

Dans notre projet, une application de télémédecine est mise en œuvre, basée sur l'acquisition du signal d'ECG d'un patient, la température ambiante, et le taux d'humidité dans l'espace où se trouve le patient, à travers une chaîne d'acquisition à base d'une carte ARDUINO UNO contenant un microcontrôleur Atmega 328 de Atmel. Puisque la cible primaire du projet est de fournir la mobilité et le confort au patient, il est nécessaire d'employer une communication sans fil puis ce que la transmission de données doit être continue, on a choisi d'utiliser la technologie Bluetooth puis qu'elle peut et facilement être reliée aux systèmes informatisés.

Notre travail est structuré comme suit :

Dans le premier chapitre nous parlerons sur l'anatomie du cœur, et la notion de la cardiologie et du système cardiovasculaire et l'électrocardiographie.

Le second chapitre va se porter sur la description du système dont la présentation de la carte ARDUINO UNO (principe de fonctionnement et avantages) et l'étude de l'étage amplificateur, aussi la description du capteur numérique SHT75, et on termine cette partie par la description du module bluetooth utilisé

Le troisième chapitre sera consacré sur la présentation des différents logiciels utilisé le long de ce travail.

Nous terminerons notre travail par une conclusion générale.

Chapitre I

Les principes de base du signal ECG

I- Préambule :

Nous présenterons dans ce premier chapitre l'anatomie et le fonctionnement du cœur ainsi que le système cardiovasculaire. Des généralités sur le signal d'ECG, l'ECG Holter et les différentes dérivations.

II-1-Anatomie et fonctionnement du cœur :

Le cœur est un organe musculaire assurant le rôle d'une pompe au niveau du système sanguin. Son activité à la fois électrique et mécanique assurant le pompage de 8000 litres de sang par jour avec 100 000 battements quotidiennement en moyenne. Il est situé au milieu du médiastin où il est partiellement recouvert par les poumons et antérieurement par le sternum et les cartilages des troisièmes, quatrièmes et cinquièmes côtes. Les deux tiers du cœur sont situés à gauche de la ligne passant par le milieu du corps. Il repose sur le diaphragme et est incliné en avant et à gauche de telle sorte que l'apex soit antérieur par rapport au reste du cœur.

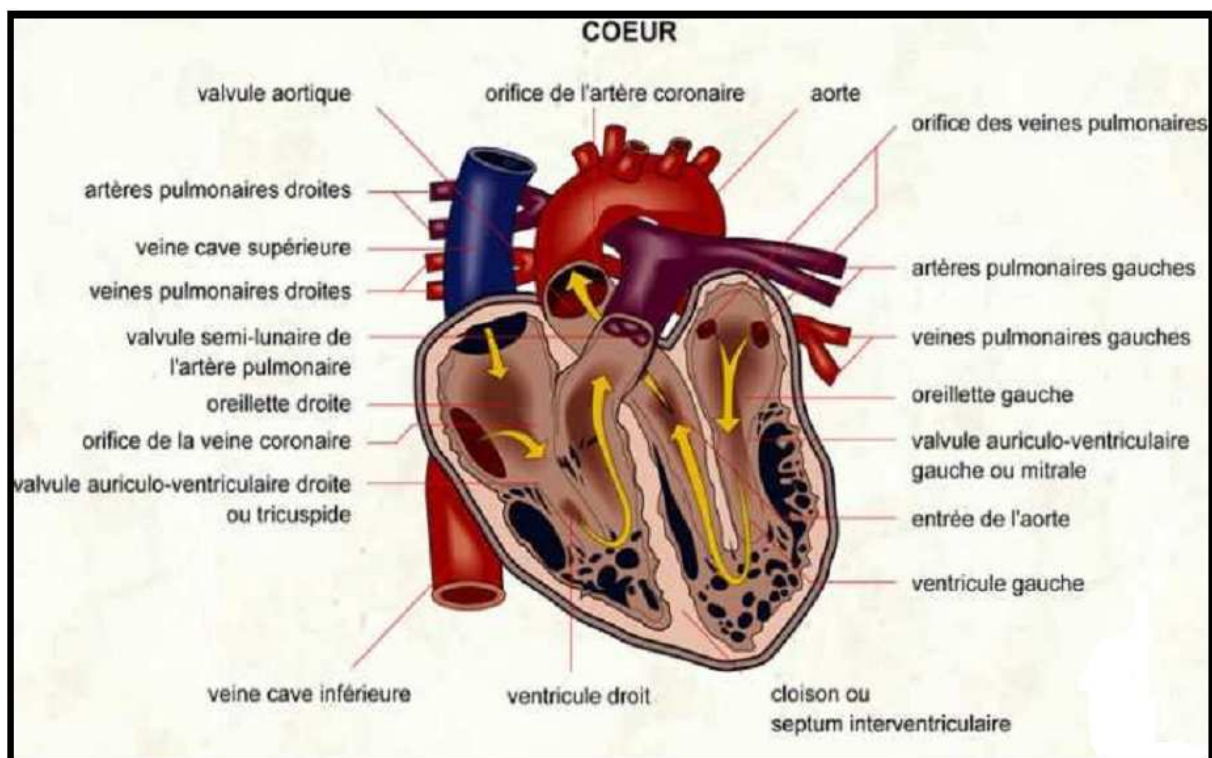


Figure 1-1 : Schéma et anatomie du cœur

Le cœur est constitué de quatre cavités contractiles: les oreillettes reçoivent le sang veineux et les ventricules droit et gauche le propulsent

respectivement dans la circulation pulmonaire et dans la circulation systémique (**Figure 1-1**).

Chaque battement cardiaque est un processus mécanique engendré par des phénomènes bioélectriques, notamment ioniques.

La paroi du cœur est formée d'un tissu musculaire contractile appelé **le myocarde**. Il existe dans ce dernier un tissu musculaire très particulier qui est **le tissu nodal**, ce tissu joue un rôle primordial, car il stimule et entretient les battements cardiaques.

Ce tissu est constitué essentiellement de :

a-Nœud sinusal : localisé dans la paroi de l'oreillette droite tout près de la veine cave supérieure, il détermine la fréquence cardiaque (pour un être humain en bonne santé est d'environ 70 battements par minute), est aussi le pacemaker pour le rythme cardiaque normal.

b-Nœud Auriculo-ventriculaire AV : il occupe la partie intérieure de la cloison inter-auriculaire.

c-Faisceau de His : ce faisceau pénètre dans la cloison inter ventriculaire et se divise en deux branches droite et gauche qui se ramifient dans le myocarde ventricule.

d-Réseau de Purkinje : ce réseau est créé par la ramification des deux branches du faisceau de His.

L'ensemble de ces éléments constitue aussi le réseau de conduction électrique du cœur, c'est-à-dire que la stimulation électrique du myocarde part du nœud sinusal (ou nœud de KEITH & FLACK), elle traverse l'oreillette et elle transite par le nœud auriculo-ventriculaire (ou nœud d'ASCHOFF-TAWARA), avant de rejoindre les ventricules via le réseau de distribution nodal c.à.d. le faisceau de HIS, les branches de TAWARA et le réseau terminal de PURKINJE.

On peut voir tout ces éléments dans la **figure1-2** :

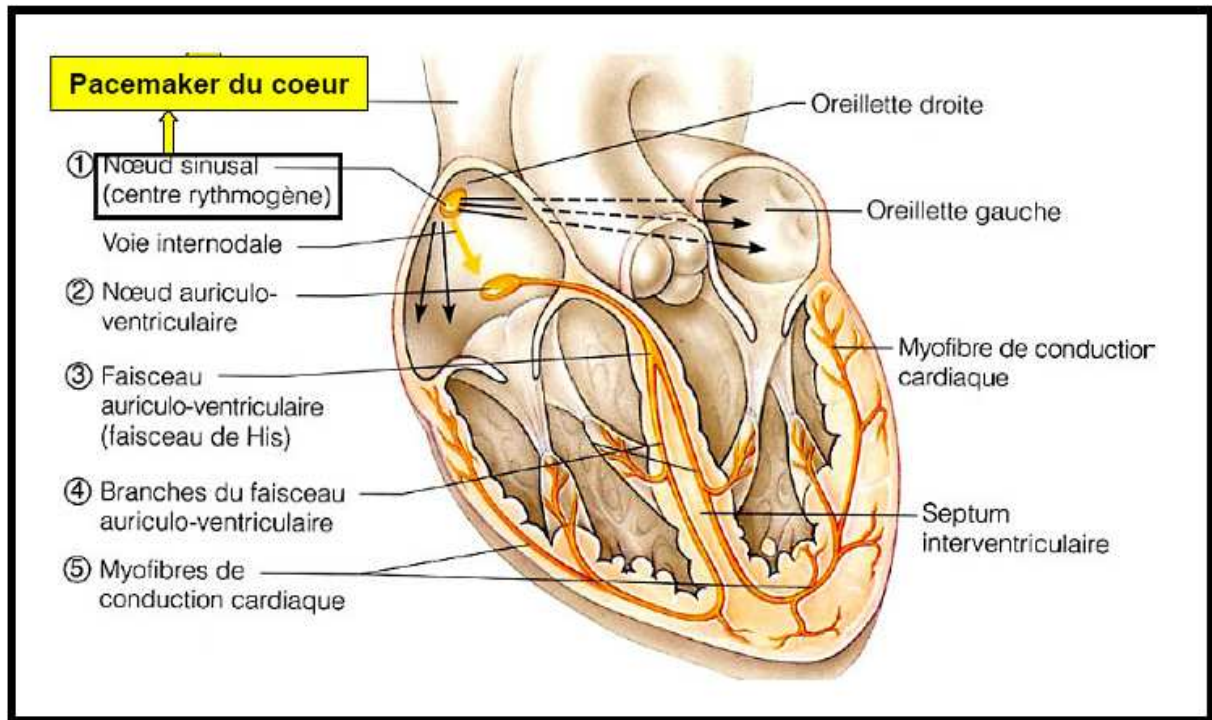


Figure1-2 : Illustration schématique du système de conduction électrique du cœur

II-2-Le cycle cardiaque :

Le cycle cardiaque consiste en trois étapes majeures : la systole auriculaire, la systole ventriculaire, et la diastole.

La systole auriculaire consiste à la contraction des oreillettes pour chasser le sang dans les ventricules en passant par une valvule à sens unique (anti retour).

La systole ventriculaire dans son sens implique la contraction des ventricules expulsant le sang vers le système circulatoire.

La troisième étape qui est la diastole consiste en elle-même à la relaxation de toutes les cavités, puis le démarrage du cycle à nouveau.

Ce cycle est commandé par des impulsions générées par les fibres du myocarde.

II-2-1-Le potentiel d'action :

Il est très important à ce niveau, d'étudier le comportement des cellules cardiaques appelées **les myocytes**. Ce sont les cellules responsables de l'activité électrique du cycle cardiaque, le potentiel dit d'action se produit comme suit :

Au début, la cellule est au repos on dit qu'elle est polarisée, le potentiel de la face interne de la cellule étant négatif par rapport à celui du milieu environnant. Cette différence de potentiel représente le potentiel de la membrane, elle est dû à l'écoulement des ions de Sodium (Na^+) et de Potassium (K^+) à travers la membrane cellulaire (**Figure 1-3**), elle est comprise entre 70 et 100 mV.

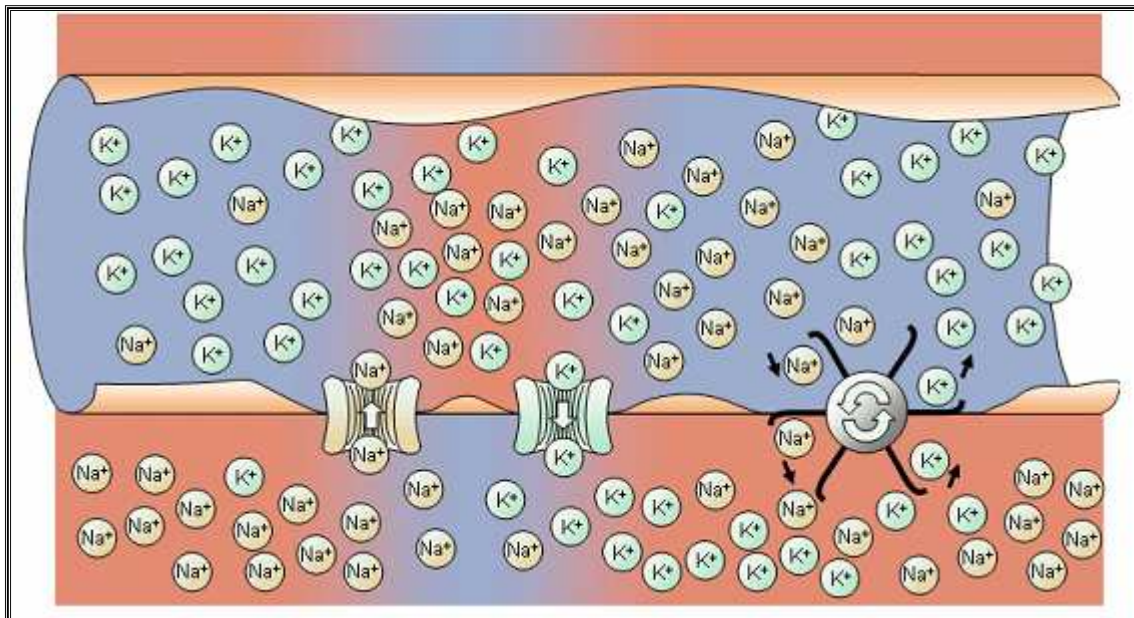


Figure 1-3 Ecoulement des ions de Potassium K^+ et de Sodium Na^+ à travers la membrane cellulaire

Si on excite la surface de la cellule avec une charge négative telle que le potentiel de la membrane s'abaisse au moins d'un tiers, une décharge (dépolarisation) de celle-ci surviendra alors spontanément. La cellule cardiaque répond à ce phénomène par une contraction. La polarité à travers la membrane cellulaire est inversée. Le potentiel de la face interne est alors approximativement de 25 mV plus élevé que celui de la surface externe de la cellule. Après la dépolarisation (phénomène initial et rapide), la cellule se repolarise instinctivement (phénomène terminal et lent) et revient à l'état de repos.

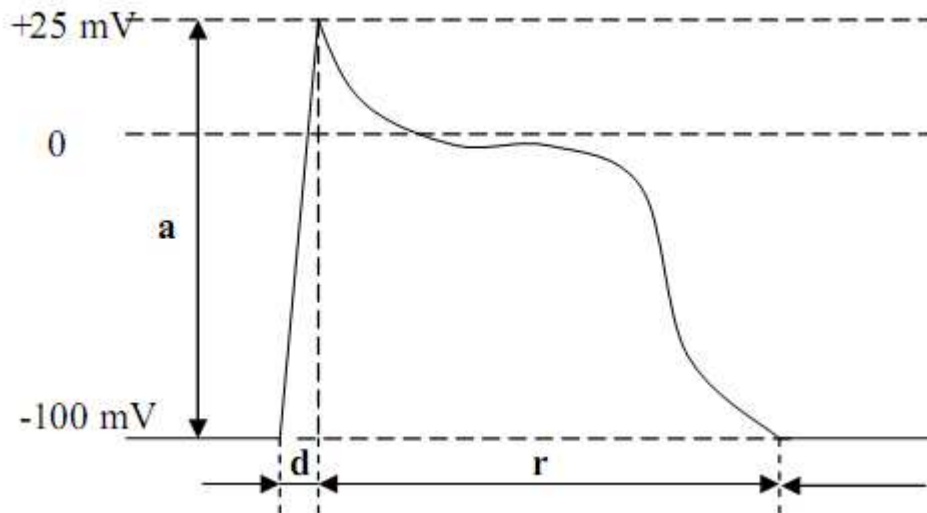


Figure 1-3 : Courbe du potentiel d'action : Dépolarisation ' d ' (rapide) et Repolarisation ' r ' (lente).

La **figure 1-3** montre la succession d'événements électriques par une courbe de variations de la différence de potentiel (**ddp**) entre les faces de la membrane cellulaire au cours d'un seul cycle.

La variation de la ddp de -100 à 25 mV s'appelle « potentiel d'action ». Lorsqu'un phénomène électrique altère suffisamment le milieu au contact d'un tissu excitable, il se produit un état de déséquilibre qui se propage de proche en proche à toutes les parties du tissu. Cette propagation s'accompagne avec une onde électrique enregistrable avec des appareils adéquats.

II-2-2-Activation cardiaque physiologique et conduction intracardiaque:

L'onde d'activation naît dans l'oreillette droite, dans le nœud sinusal, situé au pied de la veine cave supérieure (**Figure 1-4a**). Cette onde diffuse ensuite à travers les deux oreillettes et atteint le nœud auriculo-ventriculaire d'ASCHOFF-TAWARA (**Figure 1-4b**), où elle subit une conduction décrémenteille (**Figure 1-4c**). L'onde de dépolarisation subit un ralentissement à ce niveau, puis parcourt le système HIS-PURKINJE (**Figure 1-4d**).

Le tronc du faisceau de His se bifurque en deux branches droite et gauche :

- l'unique branche droite, parcourt la bandelette anciforme du ventricule droit.

- la branche gauche se divise quasi immédiatement en deux faisceaux antérieurs (fin et fragiles) et postérieur (plus épais et résistant).

À la sortie des branches du faisceau de His, l'impulsion électrique s'engage dans le myocarde ventriculaire à travers le réseau de Purkinje sous l'endocarde des deux ventricules. L'onde de dépolarisation atteint enfin le myocarde indifférencié (**Figure 1-4e**), qui se dépolarise de l'endocarde vers l'épicarde.

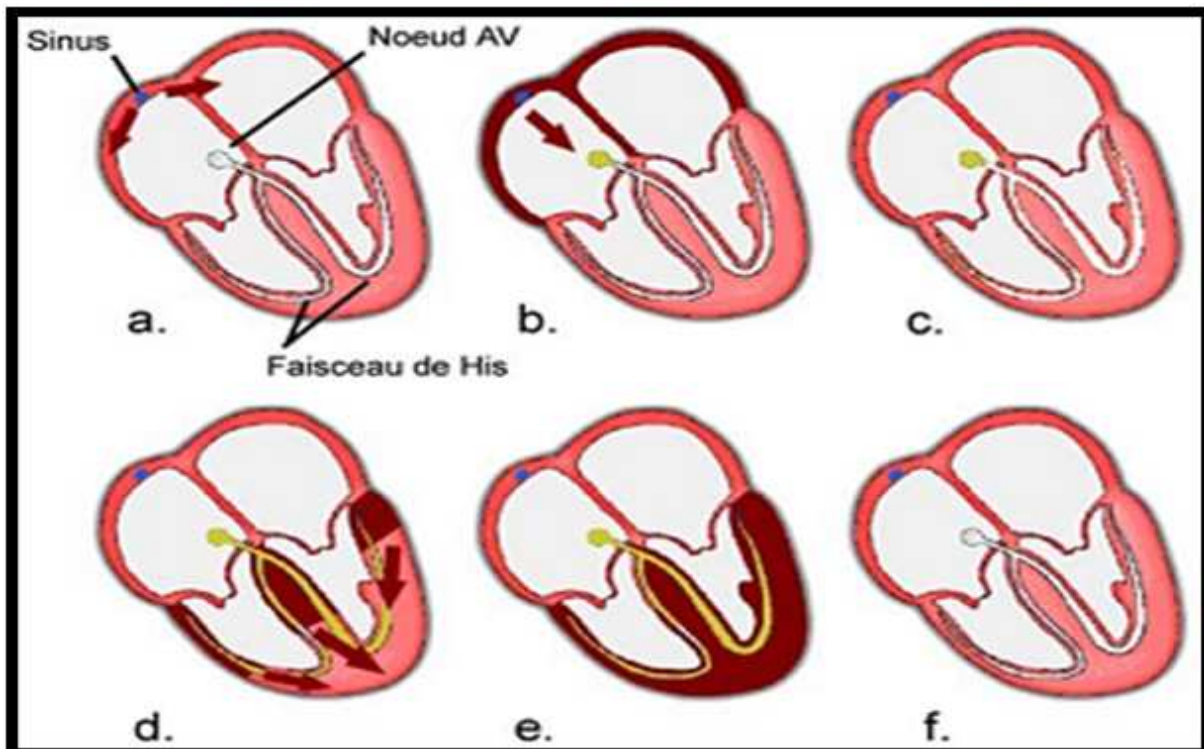


Figure 1-4 : Activation électrique et physiologique du myocarde

Le ralentissement de l'onde de dépolarisation, au niveau du nœud auriculo-ventriculaire (durée : 0,12 à 0,20s), permet aux ventricules d'être stimulés avec un certain retard par rapport aux oreillettes, ce qui favorise le remplissage ventriculaire passif qui sera complété en fin du diastole par la contraction auriculaire.

Aux étages auriculaire et ventriculaire, la conduction se fait de proche en proche, avec une direction bien établie :

- **Étage auriculaire** : l'impulsion électrique se propage de l'OD vers l'OG, à travers le septum inter-auriculaire, du plafond vers le plancher des oreillettes (durée 0,04s).

- **Étage ventriculaire** : le VD et le VG se dépolarisent au même temps (ce qui permet leurs synchronisations mécaniques) sur une durée de 0,10 s. L'activation débute au niveau de la région moyenne du septum du VG vers le VD (les premières 0,02 s) puis s'étend rapidement au reste des ventricules. La partie la plus basale des parois ventriculaires et du septum est la dernière activée (les dernières 0,02 s).

La **figure 1-5** nous permet de voir les différentes formes d'ondes qui apparaissent dans les différents tissus du muscle cardiaque :

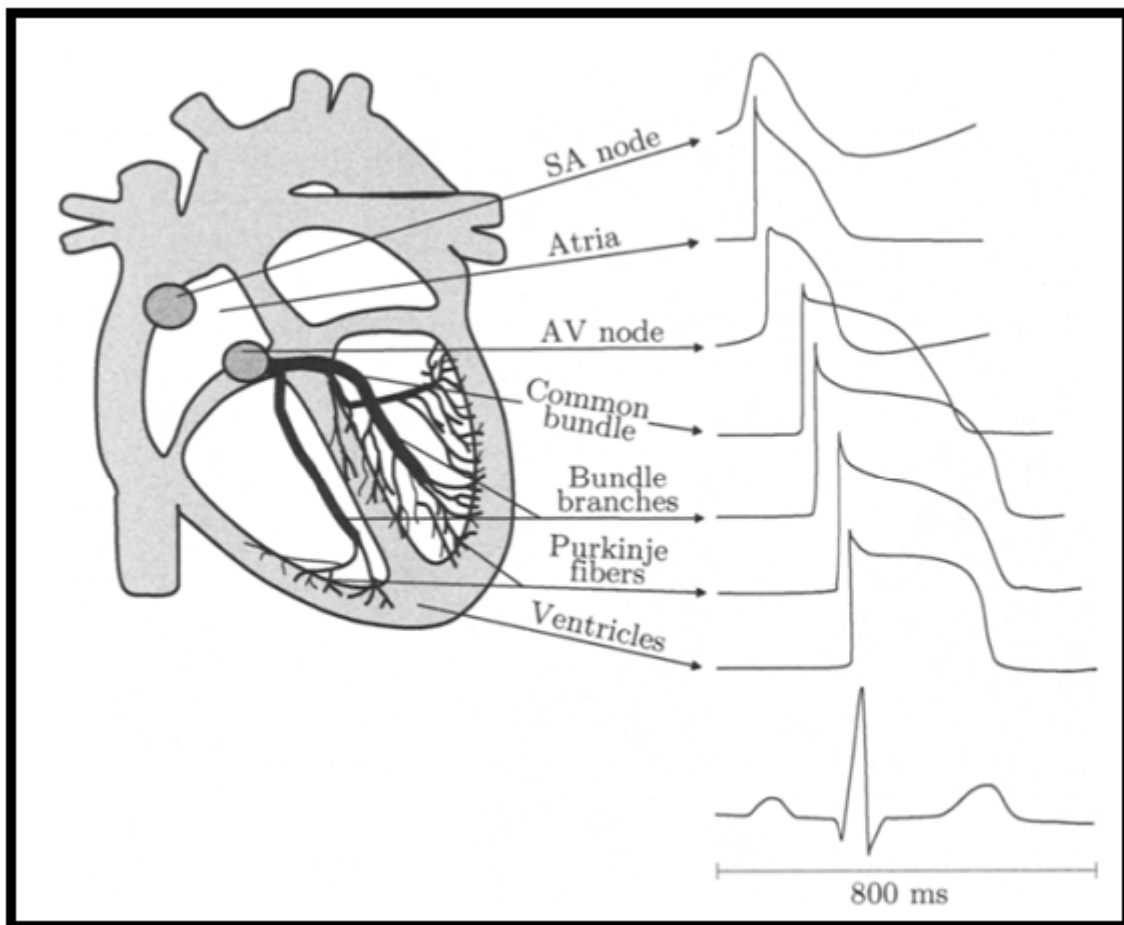


Figure1-5 Les différentes formes d'ondes qui apparaissent dans les différents tissus du cœur

III- L'électrocardiogramme :

L'électrocardiogramme abrégé par l'acronyme "**ECG**", est un signal décrivant l'activité électrique du cœur recueillie par les électrodes placées à la surface du patient. La variation de la tension mesurée par les électrodes est causée par le potentiel d'action des cellules cardiaques précédemment citées.

III-1-Historique :

En 1880 : Augustus Waller, donne naissance au premier enregistrement du signal d'ECG d'un être humain.

En 1897 : Clement Ader, adapte un système d'amplification pour mieux voir le signal de l'ECG.

En 1903 : le physiologiste Allemand **Willem Einthoven**, fabrique le premier appareil appelé **l'électrocardiographe**, qui permet d'enregistrer le signal d'ECG à base d'un galvanomètre assez sensible, il a aussi défini l'emplacement approprié des électrodes qui est fortement utilisé aujourd'hui.

L'électrocardiographe est constitué en général des éléments suivants :

- Un ensemble d'électrodes destinées à être appliquées en contact direct avec le patient.
- Un système d'amplification des signaux issus des électrodes, qui est basé généralement sur des composants de précision comme les amplificateurs d'instrumentations.
- Un appareil enregistreur.
- Un système d'enregistrement graphique.

III-2-Enregistrement de l'électrocardiogramme :

Pour pouvoir lire le signal de l'ECG, il est nécessaire d'avoir un moyen permettant d'enregistrer ce signal, et pour cela une technique est mise en œuvre, qui est l'enregistrement de ce signal sur une feuille millimétrée ou bien, visualiser ce signal sur un moniteur (oscilloscope), puis le sauvegarder sur un support de stockage, citons l'exemple des cartes mémoire ou CD. L'acquisition du signal de l'ECG se fait par des électrodes placées sur la surface du patient suivant un emplacement qui définit un type de dérivation, puis il y a un système sur lequel sont reliées ces électrodes, pour convertir le signal acquis en une

vibration d'une plume permettant de dessiner le signal sur le papier millimétré (et ça pour les anciens électrocardiographes).

Les emplacements des électrodes sur le thorax du patient sont choisis de manière à explorer la quasi totalité du champ électrique cardiaque. En cardiologie, l'examen le plus couramment pratiqué est l'ECG 12 dérivations, où le signal électrocardiographie est visualisé selon 12 axes privilégiés :

- 6 axes dans le plan frontal qui sont : **(Figure 2-1 : (b))**

Les trois dérivations bipolaires I, II, III dites dérivations de Einthoven :

- I : enregistre les différences de potentiel électrique entre le poignet droit et le poignet gauche.
- II : enregistre les différences de potentiel électrique entre le poignet droit et la jambe gauche.
- III : enregistre les différences de potentiel électrique entre le poignet gauche et la jambe gauche.

Plus les dérivations unipolaires AVR, AVL, AVF dites de Wilson

- AVL (left) pour l'avant bras gauche.
- AVR (right) pour l'avant bras droit.
- AVF (foot) pour la jambe gauche.

- 6 axes sur le plan transversal (dérivations unipolaires précordial V1 à V6 dite de Kossman) : **(Figure 2-1 : (a))**

V1 : 4ème espace intercostal droit, bord droit du sternum (parasternal).

V2 : 4ème espace intercostal gauche, bord gauche du sternum (parasternal).

V3 : à mi-chemin entre V2 et V4.

V4 : 5ème espace intercostal gauche, sur la ligne médio claviculaire.

V5 : même horizontale que V4, ligne axillaire antérieure.

V6 : même horizontale que V4, ligne axillaire moyenne.

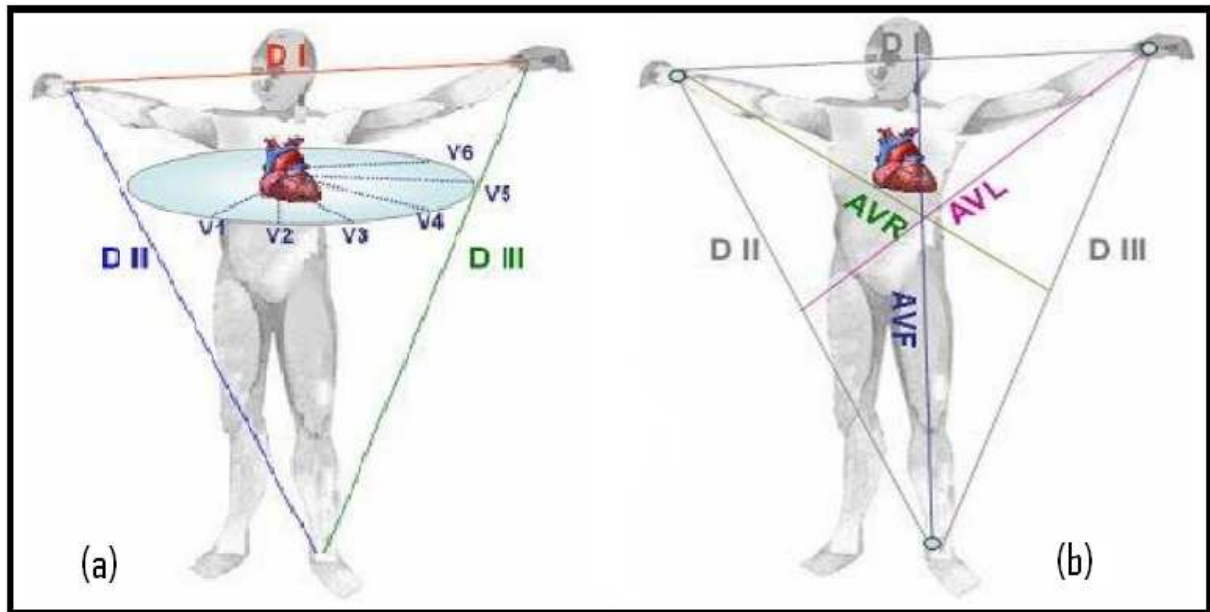


Figure 2-1 : (a) : Les dérivations précordiales. (b) : Position des électrodes des dérivations bipolaires et des dérivations unipolaires.

III-2-1-L'ECG de Holter :

Parfois, il est nécessaire de suivre l'activité cardiaque du patient pas uniquement pour un segment de temps mais pour une durée importante qui dépasse 1 heure voir des heures pour pouvoir diagnostiquer des arythmies transitoires, le diagnostic de l'insuffisance coronaire et la modification du segment ST-T (segments qu'on va voir un peut plus loin dans ce chapitre), et pour se faire, une technique appelée **le Holter ECG** est mise en œuvre.

Le Holter ECG est un examen dérivé de l'électrocardiogramme, indolore et ambulatoire, qui permet d'enregistrer l' électrocardiogramme en continu sur 24 heures consécutives, au moyen d'un boîtier relié au thorax par des électrodes (système embarqué). Technique proposée par **Norman Holter** dans les années 60.

Cet examen est particulièrement utile pour dépister des troubles du rythme cardiaque et/ou de la conduction électrique qui peuvent être à l'origine de syncopes, malaises ou palpitations.

III-3-Les électrodes :

Les électrodes sont des sondes utilisées pour recueillir des potentiels bioélectriques comme ceux provoqués par les cellules de muscle et de nerf. Les électrodes d'ECG sont généralement de type à contact direct (c.-à-d. contacte directe avec la peau). Elles fonctionnent comme capteurs convertissant l'écoulement ionique du corps à travers un électrolyte en une circulation d'électrons et par conséquent, sont capable de mesurer le potentiel électrique. Ces capteurs, connus sous le nom « électrodes enfoncées », se composent généralement d'un métal tel que l'argent ou l'acier inoxydable, avec un électrolyte de gelée qui contient le chlorure et d'autres ions (**Figure 2-2**).

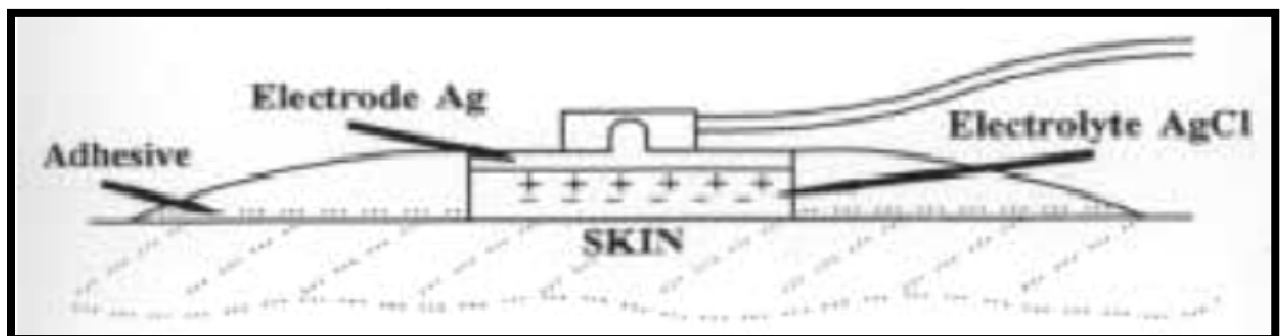


Figure 2-2 : Structure d'une électrode

III-3-1-Emplacement des électrodes :

Plus les électrodes sont proches du cœur, plus le signal qui est obtenu, est fort. L'emplacement des électrodes sur le thorax d'un patient fournit un signal plus fort que celui des poignets et des jambes. En accord avec le modèle du dipôle vectoriel du cœur et de l'équation mathématique donnant la tension différentielle obtenue comme produit scalaire du dipôle vectoriel du cœur et du vecteur entre deux électrodes, plus ces deux vecteurs sont étroits de la perpendiculaire, plus la tension obtenue est grande.

Au cours des essais qu'on a fait pour acquérir le signal d'ECG dans notre projet, on a utilisé des électrodes commerciales et des disques en métal, le résultat obtenu avec les électrodes commerciales est excellent par rapport à celui obtenu avec les disques en métal, car le signal issu des disques en métal est entaché avec du bruit et d'interférences.



Figure 2-4 : Exemple d'électrodes utilisées dans notre projet

III-4-Tracé de l'activité électrique du cœur :

La visualisation de l'activité électrique du cœur pour un seul battement, donne un tracé constitué de six ondes successives (**Figure 2-5**) : à travers cette illustration on peut constater que le comportement du cœur peut être suivi en temps réel en utilisant le signal de l'ECG. En effet, chaque phase du cycle cardiaque a une onde bien précise. Après l'impulsion initiale provenant du nœud sinusal (qui n'est pas visible sur le signal d'ECG), l'onde se propage ensuite dans les oreillettes et entraînant leurs contraction, elle laisse la trace d'une petite déflexion positive sur l'ECG qui est l'onde P (**Figure 2-5a**). L'impulsion arrive alors au nœud auriculo-ventriculaire (AV), où se produit la courte pause qui se traduit sur l'ECG par un petit segment plat PR ; puis elle emprunte les voies de conduction rapide (le faisceau de His) pour entraîner la contraction des ventricules, suivie de leur repolarisation.

Cette propagation de l'impulsion et la contraction brève et puissante de l'ensemble du muscle ventriculaire, forment sur l'ECG une succession de 3 ondes (Q, R et S) appelé complexe QRS (**Figure 2-5b-c-d**). L'onde Q est la première : c'est une onde dirigée vers le bas, qui n'est pas toujours visible sur le tracé ; la seconde est l'onde R : elle est de grande amplitude et dirigée vers le haut ; la dernière est dirigée vers le bas : c'est l'onde S.

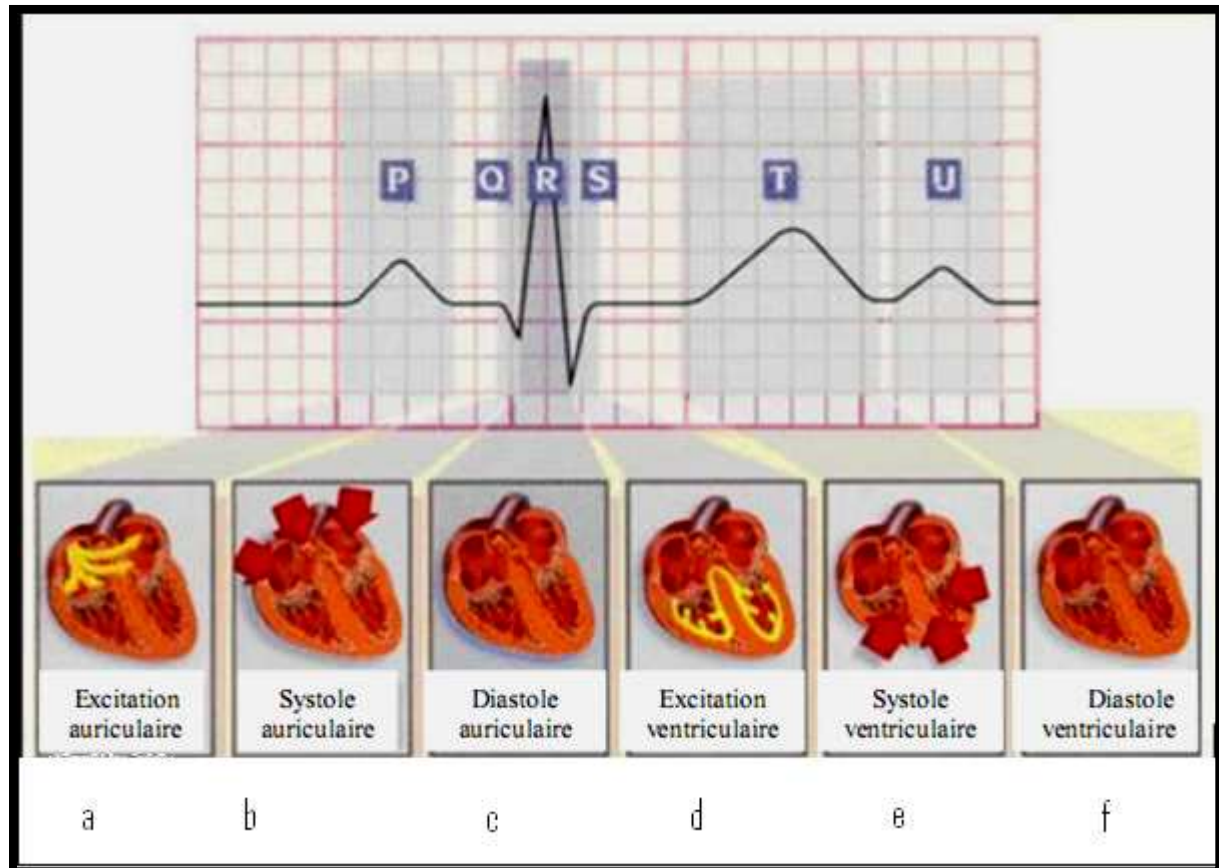


Figure 2-5 : Les étapes de reconstitution du signal d'ECG.

C'est l'ensemble de ces trois ondes qui constitue le complexe QRS. Après chaque complexe QRS, on observe sur l'ECG une onde appelée onde T (Figure 2-4e).

Entre cette onde et l'onde précédente, on note une courte pause appelée le segment ST, dont l'étude est très importante pour l'identification de certaines pathologies. L'onde T traduit la phase de repolarisation des cellules constituant les ventricules; c'est un phénomène purement électrique et pendant cette phase le cœur est mécaniquement inactif (Figure 2-4c). Enfin, le cycle se termine par une petite déflexion parfois visible qui est l'onde U (Figure 2-4f).

III-4-1-Intervalles temporels :

L'analyse de l'électrocardiogramme comprend la mesure des amplitudes et durées ainsi que l'examen de la morphologie de l'onde P, du complexe QRS, de

l'onde T, de l'intervalle PR, du segment ST, de l'intervalle QT. Les valeurs normales citées ci-dessous qui s'appliquent à des adultes d'âge moyen sont données à titre indicatif car il existe un chevauchement parfois important entre les valeurs normales et pathologiques.

- L'Intervalle PR ou PQ

L'intervalle PR ou PQ, mesuré du début de l'onde P au début du complexe QRS, représente le temps de propagation de l'influx par les oreillettes, le nœud auriculo-ventriculaire, le faisceau de His, ses branches, le réseau de Purkinje jusqu'au début de l'activation ventriculaire.

La durée de l'intervalle PR varie de 0,12 à 0,20 secondes en fonction de la fréquence cardiaque et de l'âge.

- L'Intervalle QRS

La durée de cet intervalle représente le temps de dépolarisation ventriculaire.

- L'Intervalle ST

Le segment ST est la partie du tracé comprise entre la fin du complexe QRS et le début de l'onde T. Il correspond à la phase 2, en plateau, du potentiel d'action transmembranaire. Le segment ST normal peut être légèrement décalé, vers le haut au repos, ou vers le bas à l'effort.

- L'Intervalle QT

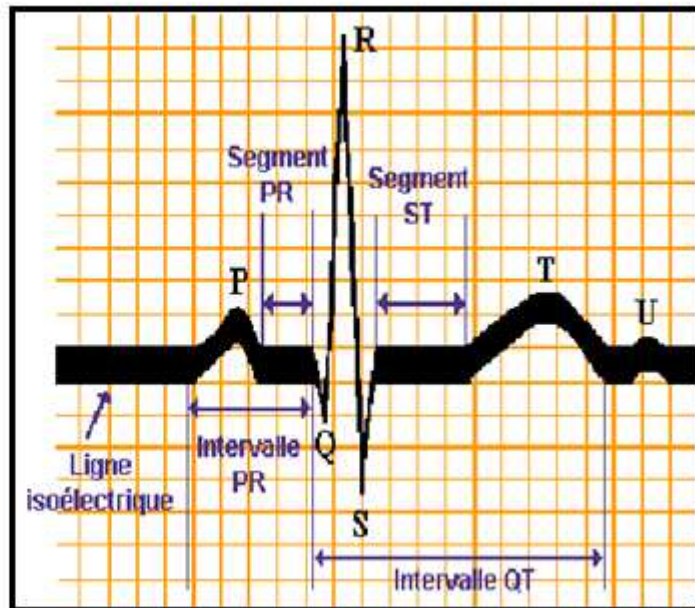
Il s'agit de la distance entre le début du complexe QRS et la fin de l'onde T, englobant la dépolarisation et la repolarisation ventriculaires. La durée de l'intervalle QT varie en fonction de la fréquence cardiaque, de l'âge et du sexe.

- L'Intervalle RR

Cet intervalle désigne le temps entre deux ondes R successives. La facilité de la détection de l'onde R donne l'importance de cet intervalle qui sert à mesurer la fréquence cardiaque, tel que $F = 1/T_{RR}$ [Hz].

-L'électrocardiogramme normal :

Figure2-6:
Figure décrivant le
signal d'ECG
normal.



IV-Conclusion :

On a vu dans ce chapitre que le signal de l'ECG est très important en terme de diagnostique et d'analyse, car il fourni toute un ensemble d'information concernant l'activité électrique du cœur, donc il est nécessaire au préalable pour réaliser un tel système d'acquisition du signal d'ECG de connaître le minimum sur le fonctionnement du cœur ainsi les différentes ondes constituant ce signal, et la durée des intervalles PQ, le complexe QRS, ST, et TU, et faire le procès des amplitudes de ces différentes ondes.

Chapitre III

Acquisition et transmission du signal ECG

I-Introduction :

Après que nous avons donné un aperçu sur la forme, la nature, et la source du signal de l'ECG, il y a eu lieu d'étudier la chaîne d'acquisition par l'intermédiaire de laquelle passe ce signal pour pouvoir le traiter ou le visualiser, ou bien le sauvegarder, et pour cela on a consacré toute cette partie pour décrire et expliquer le principe de fonctionnement de chaque étage de cette chaîne d'acquisition ainsi que tous les composants basaux utilisés.

II-Description du système :

Le système que nous avons conçu et réalisé se compose de deux parties principales : la première est la partie analogique et la seconde est la partie numérique. On commence par la partie analogique du moment que c'est elle qui s'occupe de la recueille du signal analogique, puis on passe à la partie numérique qui est réalisée à base d'une carte ARDUINO UNO.

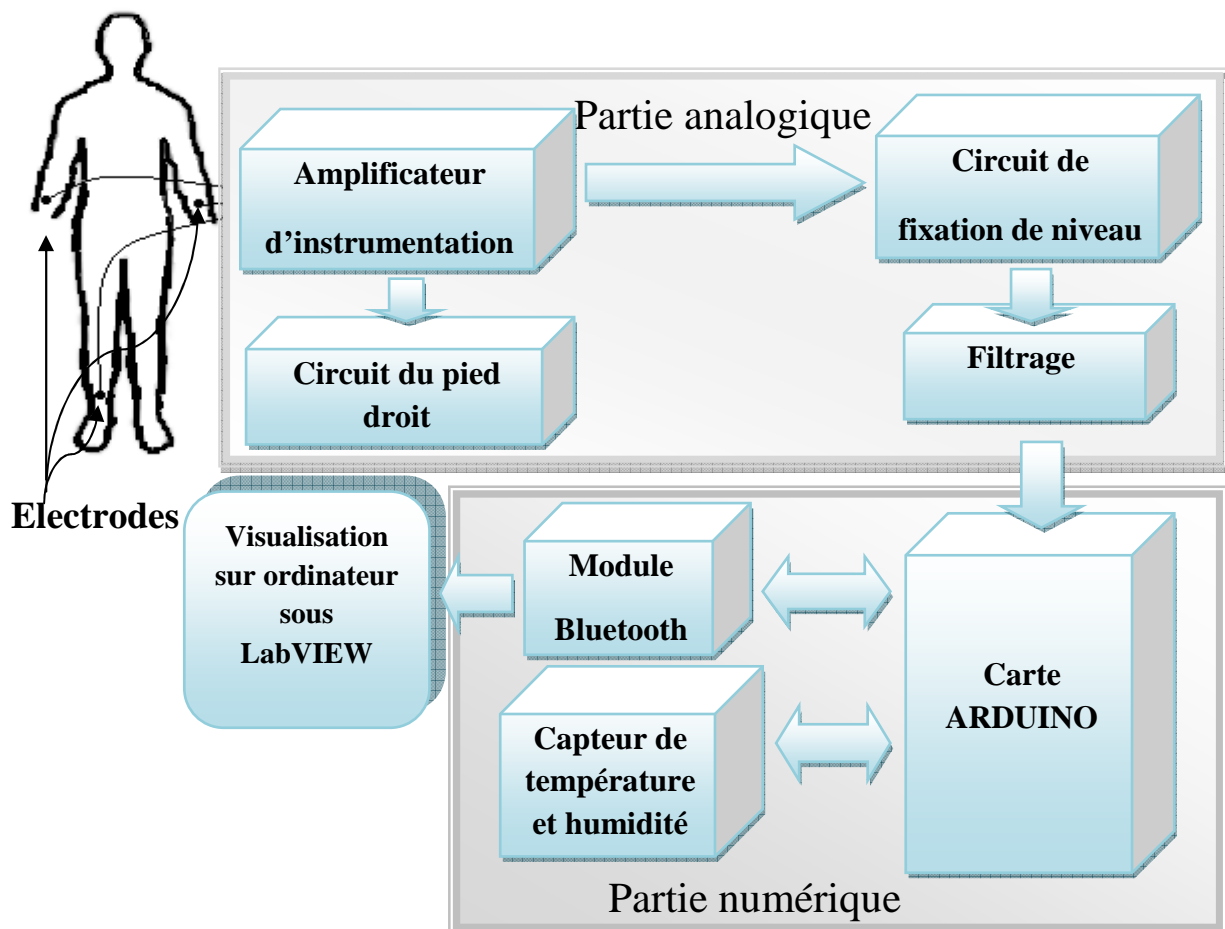


Figure 2-1 : Schéma bloqué de notre chaîne d'acquisition

La partie analogique est composée des électrodes ainsi que le circuit de conditionnement électrique (Amplification et Filtrage).

La partie numérique est composée d'une carte d'acquisition de type **Arduino Uno** basée sur un microcontrôleur **Atmega 328**, aussi c'est dans cette partie qu'on va faire inclure le capteur de température et de l'humidité du moment que c'est un capteur numérique.

Après l'acquisition de ces différentes données (le signal ECG, la valeur de la température et de l'humidité), une transmission est réalisée grâce à l'utilisation d'un module bluetooth.

II-1-L'amplificateur d'instrumentation :

Pour faire l'étude d'un phénomène physique, souvent on fait appel à un signal électrique le caractérisant, et souvent le signal il est de faible niveau donc il est nécessaire de l'amplifier, mais cette amplification ne doit concerner que le signal utile. Or ce dernier côtoie bien souvent une tension parasite (souvent du même ordre de grandeur que le signal utile) ainsi qu'une tension de mode commun dû au conditionneur associé au capteur (cas d'un pont de Wheatstone). Il faut donc faire une amplification « sélective » qui élimine ou atténue fortement tout signal ne contenant pas d'information utile pour ne garder que le signal désiré.

Pour aboutir à ce genre de problèmes on fait appel à un amplificateur dit d'instrumentation qui permet d'avoir un signal plus précis avec un important taux de rejection au mode commun.

L'amplificateur d'instrumentation est un circuit à base d'amplificateur opérationnel permettant de rejeter : des signaux inutiles, les interférences et le bruit de la tension commune des deux entées. Il est très utilisé dans le conditionnement des signaux dans les applications biomédicales, et en générale dans l'instrumentation également.

-Le taux de rejection au mode commun :

La tension de mesure (V_m) issue d'un capteur est une tension différentielle entre deux conducteurs (a et b) : $V_a - V_b = V_m$ (1)

On définit la tension du mode commun V_{mc} comme étant la tension commune à V_a et V_b et qui ne contient pas d'information. Ainsi en posant :

$$V_{mc} = \frac{V_a + V_b}{2} \quad (2)$$

On obtient ainsi les tensions : $V_a = V_{mc} + \frac{V_m}{2}$ (3)

et $V_b = V_{mc} - \frac{V_m}{2}$ (4)

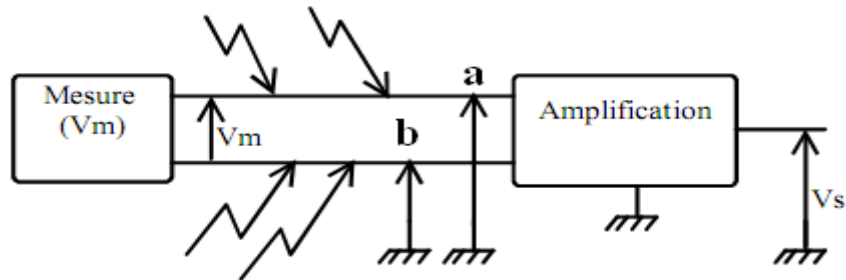


Figure 2-2 : Schéma représentant les tensions issues d'un capteur

Faire une mesure avec une bonne précision revient à rejeter la tension du mode commun V_{mc} le maximum possible de manière à extraire la tension V_m (tension différentielle de mesure) tout en étant indépendant de V_{mc} (tension de mode commun).

L'utilisation d'un amplificateur différentiel est souvent rendu nécessaire lors de la présence d'une tension de mode commun. Son rôle est de fournir en sortie, une tension proportionnelle à la différence des deux tensions d'entrée.

On peut le représenter selon la figure suivante :

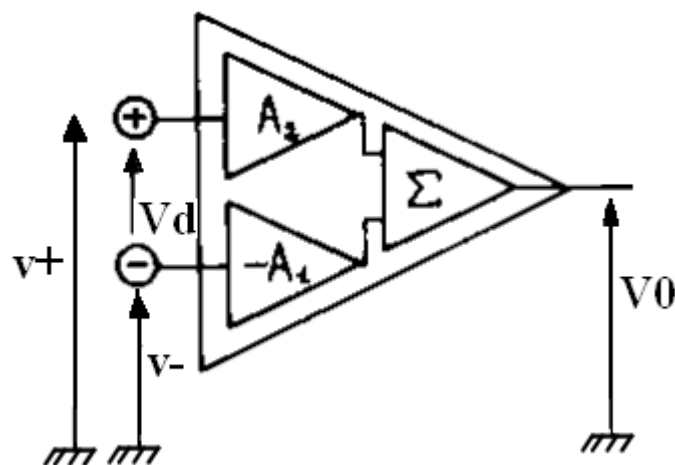


Figure 2-3 : Structure élémentaire de l'amplificateur différentiel

On dispose de deux entrées :

Entrée inverseuse de gain A_1 et Entrée non-inverseurs de gain A_2 . La sortie est un sommateur. Ainsi la tension de sortie s'écrit : $V_0 = A_2 V^+ - A_1 V^-$ (5)

En posant :
$$V_{mc} = \frac{V^+ + V^-}{2} \quad (6)$$

et
$$V_d = V^+ - V^- \quad (7)$$

On peut réécrire V_0 sous la forme :

$$V_0 = A_d V_d + A_{mc} V_{mc} \quad (8)$$

Où A_d est le gain différentiel et A_{mc} le gain de mode commun avec :

$$A_{mc} = A_2 - A_1 \quad (9)$$

et
$$A_d = \frac{A_1 + A_2}{2} \quad (10)$$

A ce niveau on définit le taux de rejection au mode commun par la formule :

$$T_{RMC} = A_d / A_{mc} \quad (11)$$

Et en décibel :
$$T_{RMC} = 20 * \log_{10} \left| A_d / A_{mc} \right| \quad (12)$$

Autant que ce paramètre est élevé autant que le signal est beaucoup plus proche du signal réel désiré, donc on doit faire tendre le gain au mode commun A_{mc} vers zéro.

- Description de l'amplificateur utilisé :

Il existe plusieurs références d'amplificateur d'instrumentation sur le marché, dans notre projet on a préféré d'utiliser le **INA114** vu sa simplicité de

mise en œuvre et son réglage simple du gain en tension en jouant sur la valeur d'une résistance appelée R_G .

La relation entre la résistance R_G et le gain en tension est caractérisé par la formule suivante :

$$G = 1 + \frac{50k\Omega}{R_G} \quad (13)$$

Donc pour avoir un gain de 1000 il suffit de donner à R_G une valeur de 50Ω , à cette valeur de gain on peut avoir un T_{RMC} d'environ 115db.

II-2-Circuit du pied droit (Righ-Leg-Drive) :

Le circuit du pied droit diffère d'un électrocardiographe à un autre, il réduit considérablement le bruit du mode commun. Dans notre projet, une électrode est reliée au pied droit du patient et elle est connectée à la sortie d'un amplificateur opérationnel auxiliaire (**Figure2-4**).

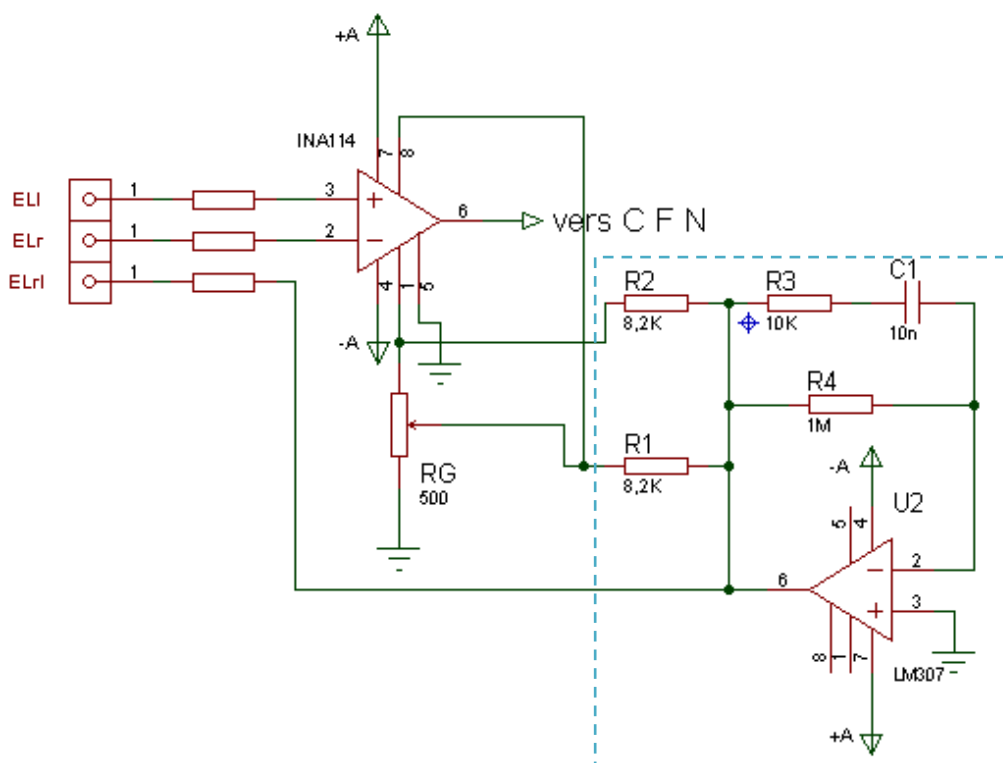


Figure 2-4 : Circuit du pied droit

II-3-Circuit de fixation de niveau CFN :

Si on prend le signal d'ECG normal on peut constaté qu'il contient des valeurs positives et des valeurs négatives comme le cas des deux ondes Q et S, mais il se trouve que les microcontrôleurs n'acceptent que des valeurs positives, donc il est obligatoire de rendre ce signal positif sans perdre son allure, la résolution de ce problème ce fait en utilisant un circuit dit de fixation de niveau permettant de décaler le signal vers l'haut (c.à.d. vers les valeurs positives), et de garder l'allure du signal telle quelle est(**Figure2-5**).

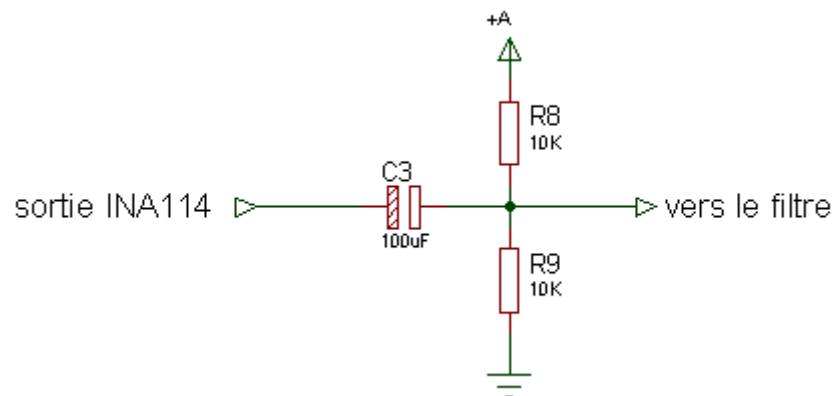


Figure 2-5 : Circuit de fixation de niveau

II-4-Le circuit de filtrage :

La fréquence d'un signal de l'ECG est comprise entre 0,3 Hz et 150 Hz environ au-delà de 150 Hz le signal de l'ECG sera mélangé avec du bruit de haute fréquence aussi en dessous de 0,3 Hz mais cette fois ci avec un bruit de basse fréquence généralement engendré par l'alimentation du secteur. Dans ces conditions le signal lui nécessite un filtrage, donc on va faire passer le signal par un circuit permettant d'enlever tous ces fréquences parasites. Du moment que notre système ne sera pas alimenté par le secteur (c.à.d. c'est un système embarqué), nous nous limitant à un filtre passe bas (du coup la réalisation d'un filtre passe bande est délicate) comme est illustré dans la figure 2-6.

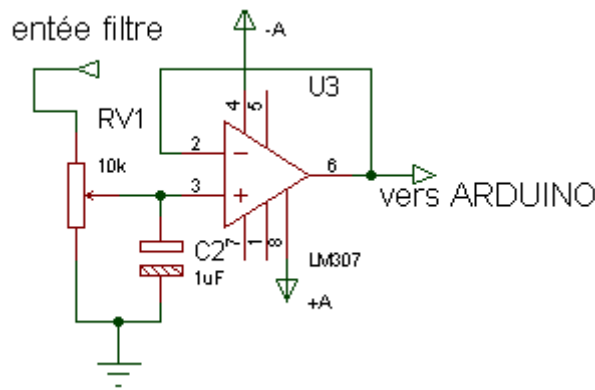


Figure 2-6 : Circuit de filtrage

La fréquence de coupure de ce filtre est obtenue à travers la formule suivante :

$$F = \frac{1}{2\pi Rv1 * C2} \text{ [Hz]} \quad (14)$$

Après que nous avons terminé la description de la partie analogique de notre système, nous passons à la seconde partie qui consiste à numériser le signal de l'ECG analogique, et mesurer la température et le taux d'humidité et transmettre le tout via Bluetooth vers un micro-ordinateur où va se faire la visualisation.

III-Description de la carte ARDUINO UNO :

Le modèle **UNO** (Figure2) de la société **ARDUINO** est une carte électronique dont le noyau est un microcontrôleur **ATMEL** de référence **ATMega328**. Le microcontrôleur **ATMega328** est un microcontrôleur 8bits 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 **ATMega8** (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 pour commencer.

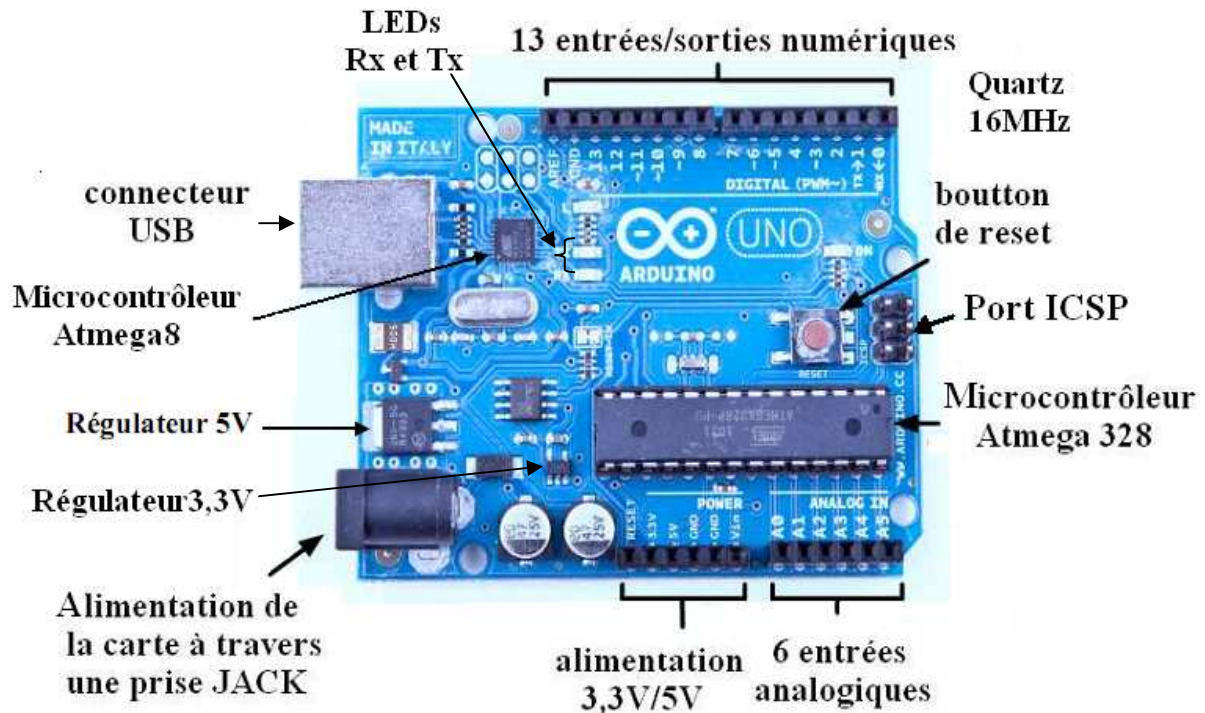


Figure 2-7 : Description de la carte ARDUINO UNO

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 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 ...

III-2-Alimentation de la carte Arduino :

La carte Arduino Uno peut-être alimentée soit via la connexion USB (qui fournit 5V jusqu'à 500mA) 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 500mA) ou des piles (ou des accus).

L'adaptateur secteur peut être connecté en branchant une prise de 2.1mm, dédié 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 7V, la broche 5V pourrait fournir moins de 5V et la carte pourrait être instable. Si on utilise plus de 12V, 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 de destruction de la carte).

Les broches d'alimentation sont les suivantes :

- **VIN** : (à distinguer du 5V 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 composant appelé un régulateur et qui est 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 50mA.

III-2-1-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.

III-3-Gestion des mémoires dans la carte ARDUINO :

L'ATmega 328 a 32Ko de mémoire FLASH pour stocker le programme (dont 0.5Ko également utilisés par le bootloader).L'ATmega 328 a également 2ko de mémoire SRAM (volatile) et 1Ko d'EEPROM (non volatile - mémoire qui peut être lue à l'aide de **la librairie EEPROM**).

III-4-Les entrées/ sorties numériques :

Chacune des 14 broches numériques de la carte UNO (numérotées des 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 5V. 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 ATmega8U2 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 a travers ces broches avec l'instruction **Serial.print()**, à condition que le câble USB soit déconnecté, sinon il va y avoir un chevauchement.

- ✓ **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. Voir l'instruction **attachInterrupt()** pour plus de détails.
- ✓ **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 librairie 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 librairie **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.

III-5-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 très utile 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 plage de mesure en utilisant la broche AREF et l'instruction **analogReference ()** du langage Arduino.

NB : les broches analogiques peuvent être utilisées en tant que broches numériques : elles sont numérotées en tant que broches numériques de 14 à 19, aux cas où le nombre de broches numériques n'est suffisant.

-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 ()**. Elle s'utilise pour réduire
- ✓ **Reset** : Mettre cette broche au niveau BAS entraîne la réinitialisation (= le redémarrage) du microcontrôleur. Typiquement, cette broche est utilisée pour ajouter un bouton de réinitialisation sur le circuit qui bloque celui présent sur la carte.

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

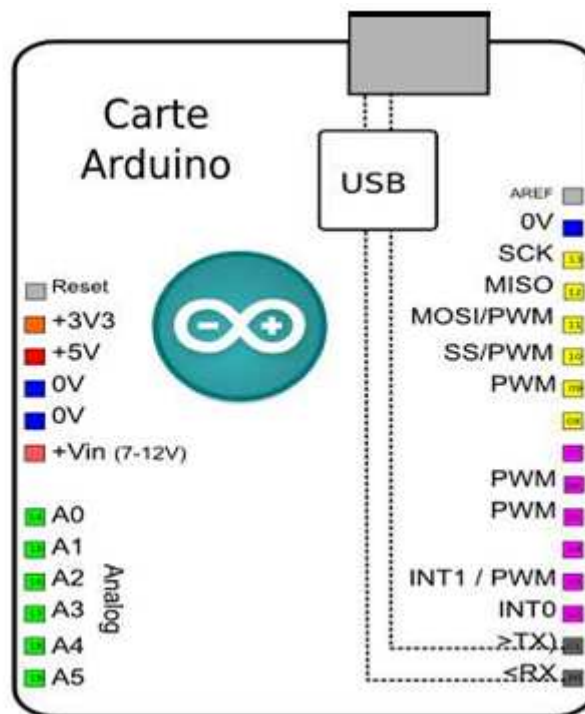


Figure 2-8 : Brochage de la carte ARDUINO UNO.

III-6-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'ATmega 328 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 (dont l'on va parler dans le chapitre suivant) 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'ATmega 328 supporte également la communication par protocole 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.

-Dimensions 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.

IV- Le microcontrôleur ATmega328 :

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

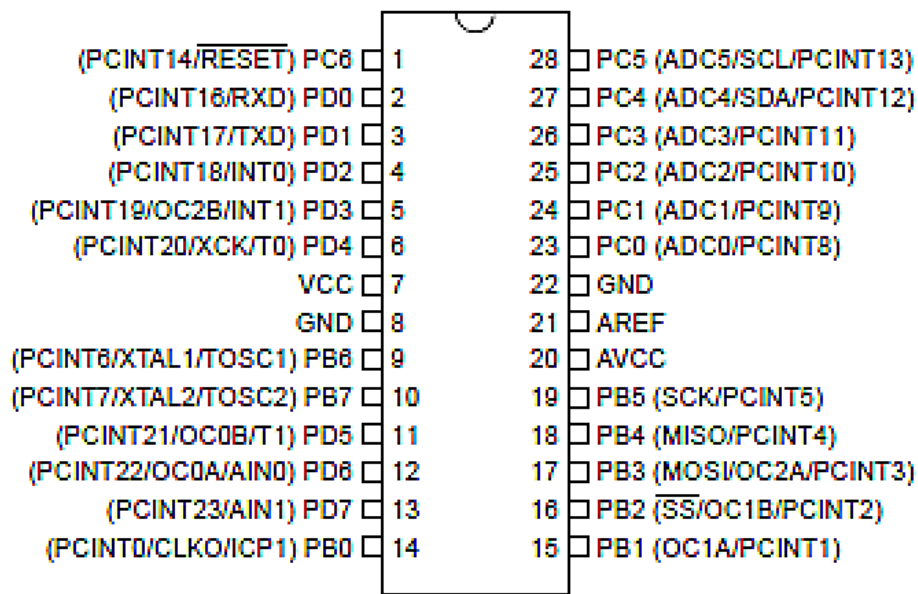


Figure 2-9 : Le microcontrôleur ATmega328.

IV-1-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 : PortB, PortC, PortD (soit 23 broches E/S en tout).
- D'un courant max par broches E /S = 40mA.
- D'un courant max sur sortie 3,3V = 50mA.
- D'une mémoire Flash de 32KB dont 512B utilisée par le bootloader.
- D'une mémoire SRAM de 2KB.
- D'une mémoire EEPROM de 1KB.

Il contient aussi trois compteurs (Timer0, Timer1, Timer2), le Timer0 et le Timer2 sont à comptage 8 bits, le Timer1 il est à comptage 16 bits. Chaque Timer peut être utilisé pour générer deux signaux PWM. Certaines broches peuvent avoir plusieurs fonctions différentes choisies par programmation :

PWM = pour l'utilisation de la 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 10bits, ce convertisseur peut être utilisé à travers 6 entrées multiplexées de ADC0(PC0) jusqu'à ADC5(PC5).

Gestion 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 le 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'interruptions (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 (plusieurs causes configurables).
- Interruption liée au comparateur analogique.
- Interruption de fin de conversion **ADC**.
- Interruptions du port série **USAR**.
- Interruption du bus **I2C**.

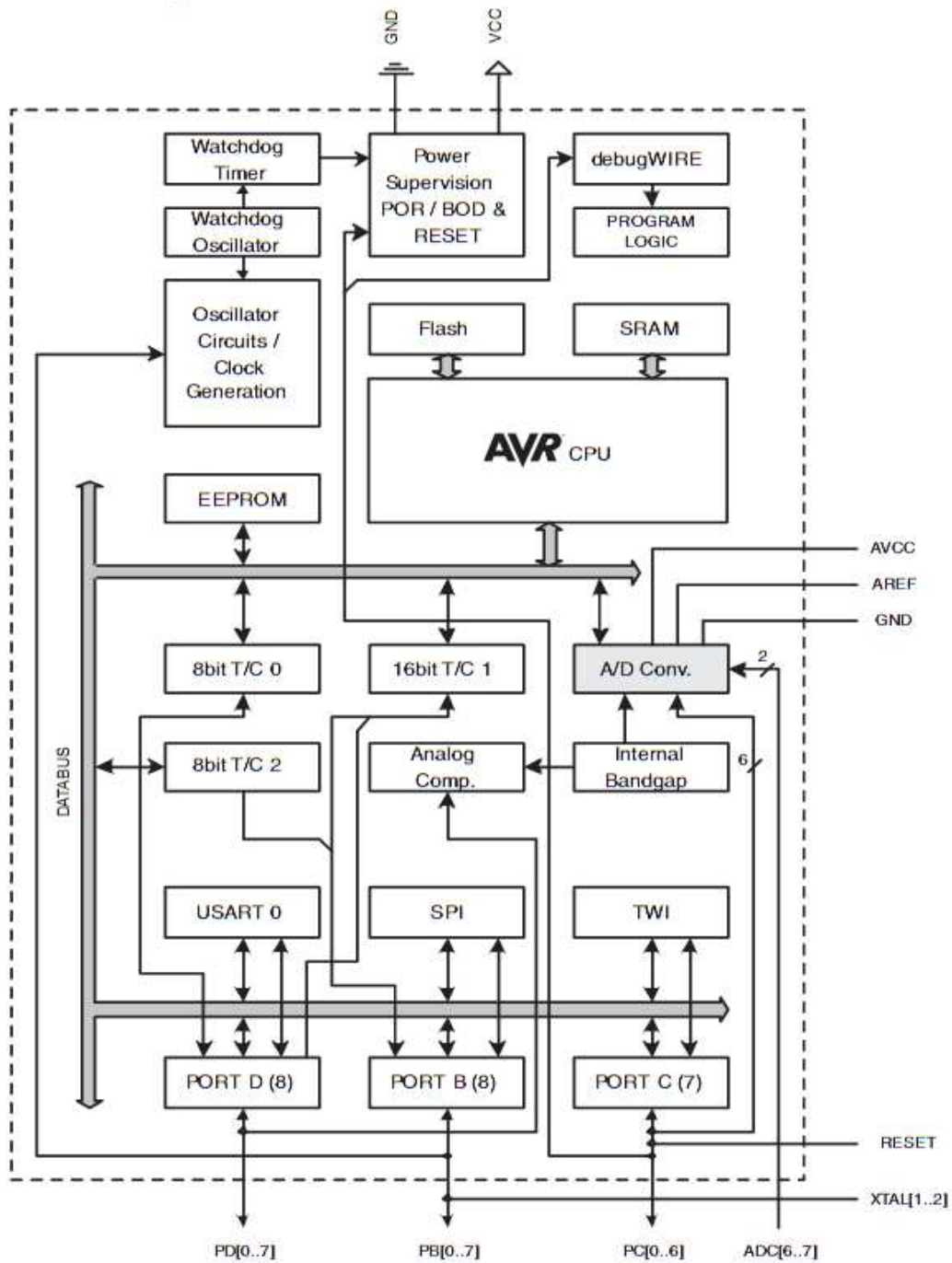


Figure 2-10 : Architecture interne de l'ATmega328

V-Le capteur de température /humidité SHT75 :

Notre problématique s'appuie sur l'influence de la température et de l'humidité sur le comportement du cœur qu'on va essayer de voir sur le signal de l'ECG, et pour cela on a choisi un capteur qui fait la mesure des deux grandeurs (température et humidité).

Le SHT75 est un module de sonde d'humidité relative et de température, comportant un résultat numérique calibré, réalisé dans la technologie CMOS, le SHT75 assure une fiabilité plus élevée et une excellente stabilité à long terme.

Le dispositif inclut deux micro-capteurs calibrés pour l'humidité relative et la température qui sont couplés à un convertisseur **analogique/ numérique** d'une résolution de 14bit, avec un circuit d'interface série. Ceci a comme conséquence : la qualité supérieure de signal, un temps de réponse réduit et l'insensibilité aux perturbations externes à un prix très concurrentiel.

L'interface série à 2 fils et la régulation interne de tension permet l'intégration facile et rapide du système. Sa taille minuscule et basses consommation de puissance, le rendent comme un dispositif meilleur trop demandé pour les applications les plus exigeantes comprenant l'instrumentation, les équipements médicaux, chauffage, dispositifs de ventilation et de climatisation, électronique grand public portable ...

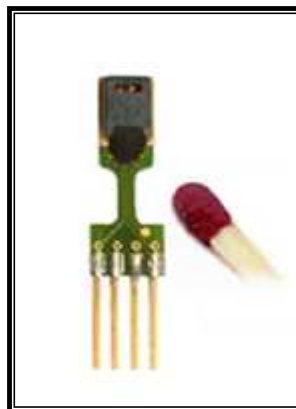


Figure 2-11 : Le capteur SHT75

V-1- Les caractéristiques du SHT75 :

Le SHT75 possède plusieurs caractéristiques spécifiques pour les deux grandeurs : humidité et température. Ces différentes caractéristiques sont décrites dans le tableau ci-dessous :

	Tolérance	Temps de réponse	Gamme
Humidité	$\pm 2,0 \%$	4S	De 0 à 100%
Température	$\pm 0,5^{\circ}\text{C}$	Entre 5 et 30S	De -40 à $123,8^{\circ}\text{C}$

Figure 2-12 : Les principales caractéristiques du SHT75.

Il faut signaler que les valeurs de la tolérance changent en fonction de la température et du taux d'humidité relative, suivant une certaine allure comme il est illustré dans la figure suivante :

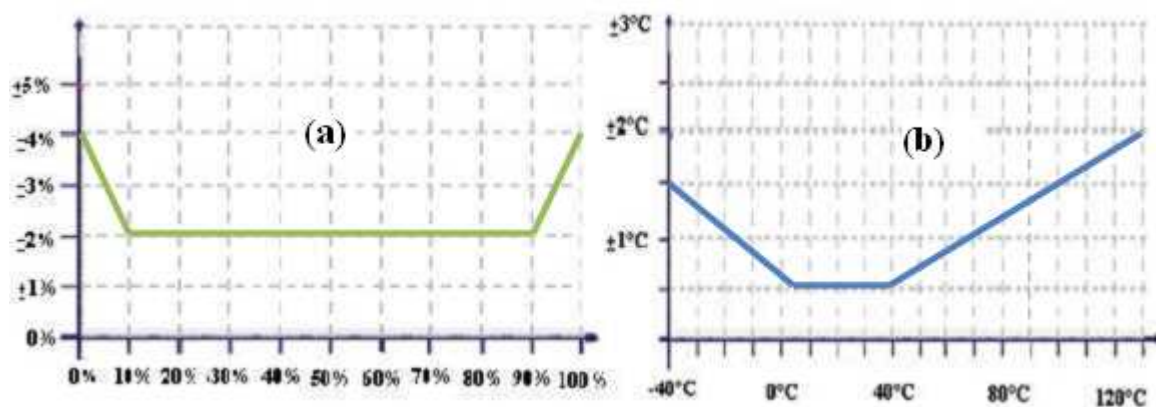


Figure 2-13 : variation de la tolérance : (a) pour l'humidité, (b) pour la température.

V-2- La structure interne du capteur :

Le SHT75 dispose de deux capteurs (l'un pour l'humidité et l'autre pour la température) miniaturisés (**Figure 2**), après juste ces deux capteurs on trouve un étage amplificateur permettant d'amplifier le signal acquis par les deux capteurs, et ensuite le signal amplifié passe par un convertisseur analogique/numérique d'une résolution de 14 bits, et il dispose aussi d'une mémoire volatile, et une interface numérique à deux fils qui permet la communication avec le microcontrôleur.

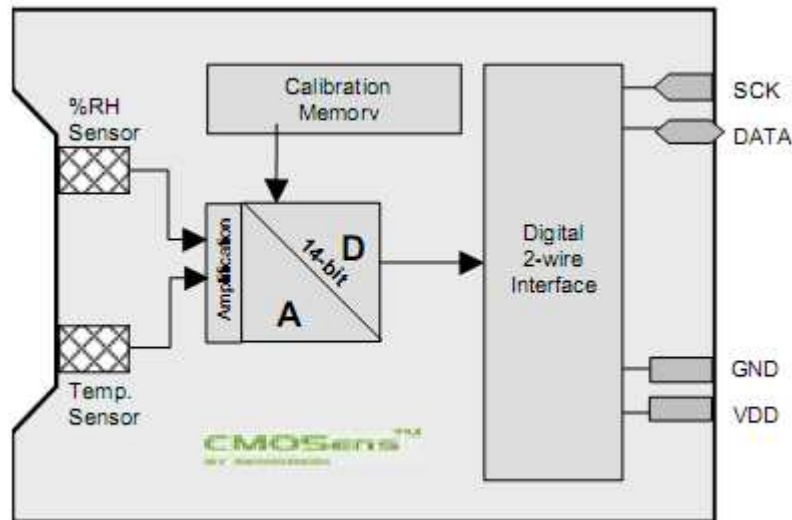


Figure 2-14 : Structure interne du SHT75.

V-3-Conversion de la sortie numérique en valeurs physiques :

Les valeurs issues du **SHT75** ce sont des valeurs numériques, mais il se trouve qu'il faut convertir ces valeurs numériques en valeurs physiques au niveau de la programmation pour qu'on puisse voir les valeurs réelles de la mesure, et pour se faire deux formules sont mises en œuvre :

1- Humidité : la formule permettant de compenser la non-linéarité du capteur d'humidité est la suivante :

$$RH_{\text{linéair}} = c_1 + c_2 * SO_{RH} + c_3 * SO_{RH}^2 \quad (15)$$

$$c_1 = -4 \quad c_2 = 0,0405 \quad c_3 = -2,8 * 10^{-6} \quad \text{pour un } SO_{RH} \text{ de 12 bits.}$$

$$c_1 = -4 \quad c_2 = 0,648 \quad c_3 = -7,2 * 10^{-4} \quad \text{pour un } SO_{RH} \text{ de 8 bits.}$$

Où SO_{RH} (*Sensor Output for Relative Humidity*) représente la valeur lue à la sortie du capteur pour l'humidité relative, donc cette valeur ensuite sera convertie à l'aide de la formule précédemment citée en une valeur physique significative.

La courbe présentée dans la **Figure 2** permet de voir les différentes valeurs du SO_{RH} pour une sortie codée sur 12 bits :

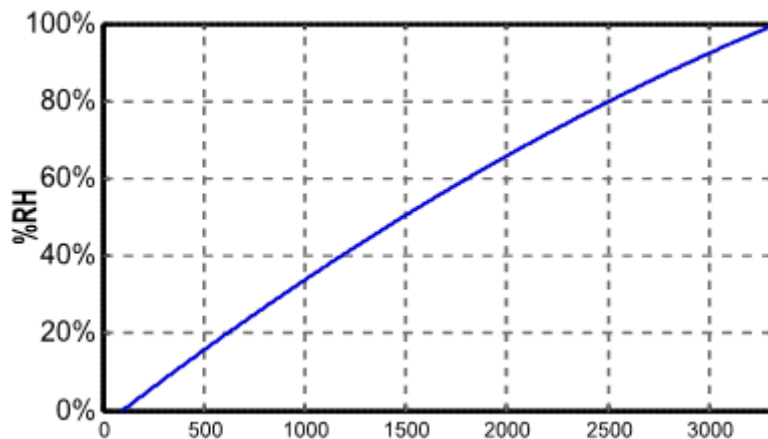


Figure 2-15 : Les valeurs du SO_{RH} codées sur 12 bits.

Pour les températures sensiblement différentes de 25°C ($\sim 77^{\circ}\text{F}$) le coefficient de la température du capteur d'humidité devrait être considéré. Dans ces conditions le taux d'humidité relative réel peut se calculé à l'aide de la formule suivante qui fait inclure le $RH_{\text{linéair}}$:

$$RH_{\text{réel}} = (T_c - 25) * (t_1 + t_2 * SO_{RH}) + RH_{\text{linéair}} \quad (16)$$

Où $t_1 = 0,01$ $t_2 = 0,00008$ pour un SO_{RH} codé sur 12 bits.

Ou bien $t_1 = 0,01$ $t_2 = 0,00128$ pour un SO_{RH} codé sur 8 bits.

2- Température : en ce qui concerne la température, la conversion des valeurs brutes issus du capteur est réalisée par une formule beaucoup moins compliquée par rapport à celle de l'humidité, cette formule est la suivante :

$$T = d_1 + d_2 * SO_T \quad (17)$$

Où SO_T (*Sensor Output for Temperature*) représente les valeurs les valeurs numériques lue à la sortie du capteur de température, et d_1 , d_2 ce sont des constantes sui dépendent de la tension d'alimentation du capteur et du nombre de bits sur lequel est codée la valeur numérique issu du capteur. Le tableau suivant permet de voir les possibles pour les constantes d_1 , d_2 :

SO _T	Celsius		Fahrenheit	
	d ₁	d ₂	d ₁	d ₂
14bit 5V	-40	0.01	-40	0.018
12bit 5V	-40	0.04	-40	0.072
14bit 3V	-38.4	0.0098	-37.1	0.0176
12bit 3V	-38.4	0.0392	-37.1	0.0704

Figure 2-16 : Les valeurs associées aux constantes d₁, d₂.

V-4- Description du SHT75 :

Le SHT75 dispose de quatre broches : deux broches (2 et 3) pour l'alimentation, une broche pour la transmission de données (1), et la dernière (4) pour le signal d'horloge :

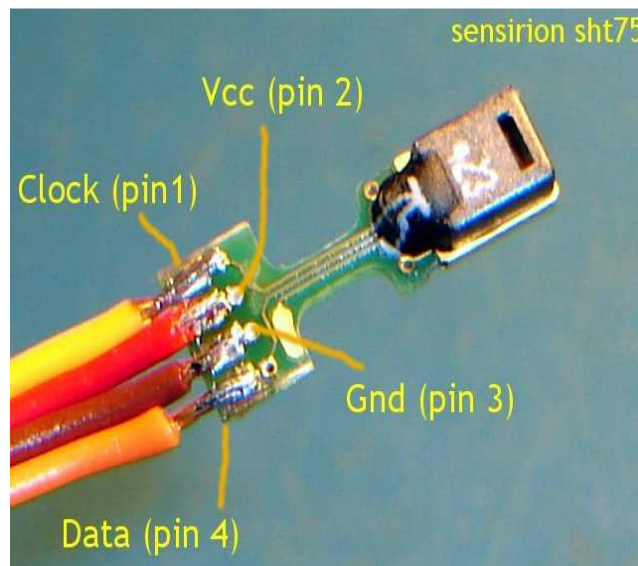


Figure 2-17 : Brochage du SHT75.

- **L'alimentation du SHT75 :**

Le SHT75 peut être alimenté avec une tension comprise entre 2,4 et 5,5V (en dessous de 2,4V le capteur ne fonctionne pas et au-delà de 5,5V c'est la destruction du capteur). Après la mise du capteur sous tension une durée de 11ms est lui nécessaire pour qu'il s'éveille, Aucune commande ne devrait être envoyée avant l'expiration de cette durée. Les broches d'alimentation (VDD, Gnd) peuvent être découplées avec un condensateur de 100 nF.

- **Les entrés/sortie :**

Le SHT75 dispose de deux broches d'entrés/sortie, une travaille comme une entrée d'horloge (**SCK : Serial Clock**), et l'autre travaille comme un bus bidirectionnel pour la transmission de données (**DATA**), ces deux broches ont plusieurs caractéristiques, on a assemblé tout ces caractéristiques dans un petit tableau qui est le suivant :

	Parameter	Conditions	Min	Typ.	Max.	Unit
F _{SCK}	SCK frequency	VDD > 4.5 V			10	MHz
		VDD < 4.5 V			1	MHz
T _{CLH}	SCK high time		100			ns
T _{CLL}	SCK low time		100			ns
T _V	DATA valid time			250		ns
T _{SU}	DATA set up time		100			ns
T _{HO}	DATA hold time		0	10		ns
T _R /T _F	SCK rise/fall time				200	ns

Figure 2-18 : Les différentes caractéristiques des signaux des entrés/sortie.

Tel que :

F_{SCK} : fréquence du signal d'horloge

T_{CLH} : temps du niveau haut.

T_{CLL} : temps du niveau bas.

T_V : temps valide de la donnée.

T_{SU} : temps de transfert de la donnée.

T_{HO} : temps de traitement ou de prise en compte de la donnée.

T_R/T_F : temps de montée et temps de descente respectivement.

La succession de ces différents temps est illustré dans les deux chronogrammes suivants :

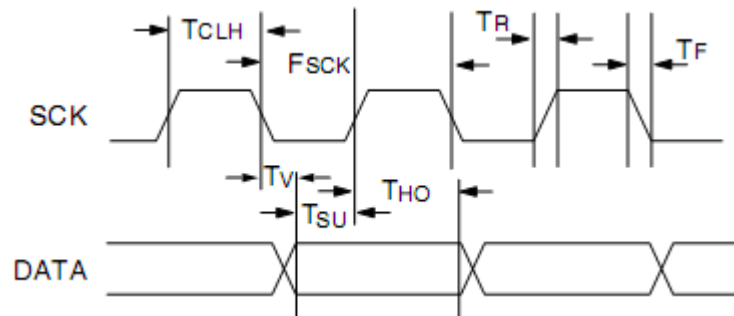


Figure 2-19 : Chronogramme du signal d’horloge et du signal de données.

Le signal d’horloge est utilisé pour synchroniser la communication entre le maître qui est le microcontrôleur et l’esclave qui est le SHT75. Les données changes au front montant et elles sont validées au front descendant bien sûr du signal d’horloge. Une résistance de rappel (pull-up) est nécessaire pour avoir l’état haut du signal. Un exemple pratique de la manière avec laquelle on relie le SHT75 à un microcontrôleur et l’emplacement de la résistance de rappel est illustré dans la **Figure 2-20** :

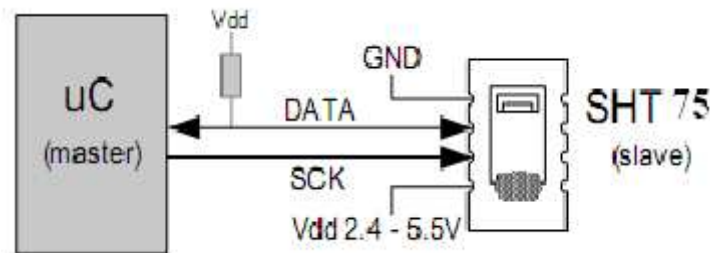


Figure 2-20 : Circuit d’application du SHT75.

V-5- La communication avec le SHT75 :

Après la mise sous tension du SHT75 et après une attente de 11ms, le processus de la détection de la température de l’humidité commence mais il se trouve qu’on ne peut pas lire directement la valeur mesuré. Deux problèmes se posent :

- 1) il y a deux mesures de deux grandeurs différentes (humidité et température).
- 2) la transmission de ces deux mesures se fait en série.

Pour aboutir à ces deux problèmes, le constructeur de ce capteur à donné une solution simple est efficace. Cette solution consiste à maitre en premier

temps le capteur comme récepteur par l'envoi d'une succession de bits significatifs et ensuite le transformer en émetteur pour qu'il puisse envoyer la valeur d'une mesure. C'est ce processus qui se répète pour la mesure des deux grandeurs.

Pour initialiser la transmission, un bit dit d'initialisation devrait être transmis vers le capteur. Il consiste à envoyer un bit nul au capteur quand le signal d'horloge est au niveau haut suivi d'un niveau bas (de l'horloge) d'une durée inférieure à celle du niveau haut précédent et de faire basculer le signal de données vers le niveau haut tendit que le signal d'horloge est encore au niveau haut. Les chronogrammes suivants permettent de comprendre cette succession de bits :

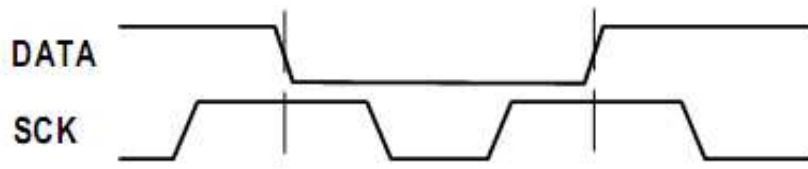


Figure 2-21 : Chronogrammes d'initialisation de la transmission

La suite de séquence après l'initialisation de la transmission c'est une succession de trois bits dit d'adresse puis cinq bits de commande (**la figure 2** nous permet de voir ces différents codes de commande). Le SHT75 indique la bonne réception de la donnée après une petite durée d'attente, par la mise au niveau bas du signal de données qui s'appelle bit de reconnaissance **ACK bit (acknowledge bit)**, ce bit est envoyé après le huitième front descendant du signal d'horloge, ensuite la ligne de transmission se libère après le neuvième front descendant du même signal d'horloge.

Command	Code	Description
Reserved	0000x	Reserved
Measure Temperature	00011	Temperature measurement
Measure Humidity	00101	Humidity measurement
Status Register Read	00111	Read access to the status register (see application note)
Status Register Write	00110	Write access to the status register (see application note)
Reserved	0101x-1110x	Reserved
Soft reset	11110	resets the chip, clears the status register to default values wait 11ms before next command

Figure 2-22 : Les différents codes de commande du SHT75.

V-5-1 La mesure de l'humidité et de la température :

Les codes de la commande du SHT75 pour qu'il mesure la température et l'humidité sont "00000011" et "00000101" respectivement, le microcontrôleur doit attendre jusqu'à ce que la mesure soit complète. La durée nécessaire est approximativement : 11, 55, 210ms pour une mesure qui se fait sur 8, 12, 14bits respectivement. La durée exacte varie de $\pm 15\%$ des valeurs précédemment citées avec la fréquence de l'oscillateur interne. Le SHT75 signale la fin de la mesure par la mise au niveau bas du signal de la ligne de données.

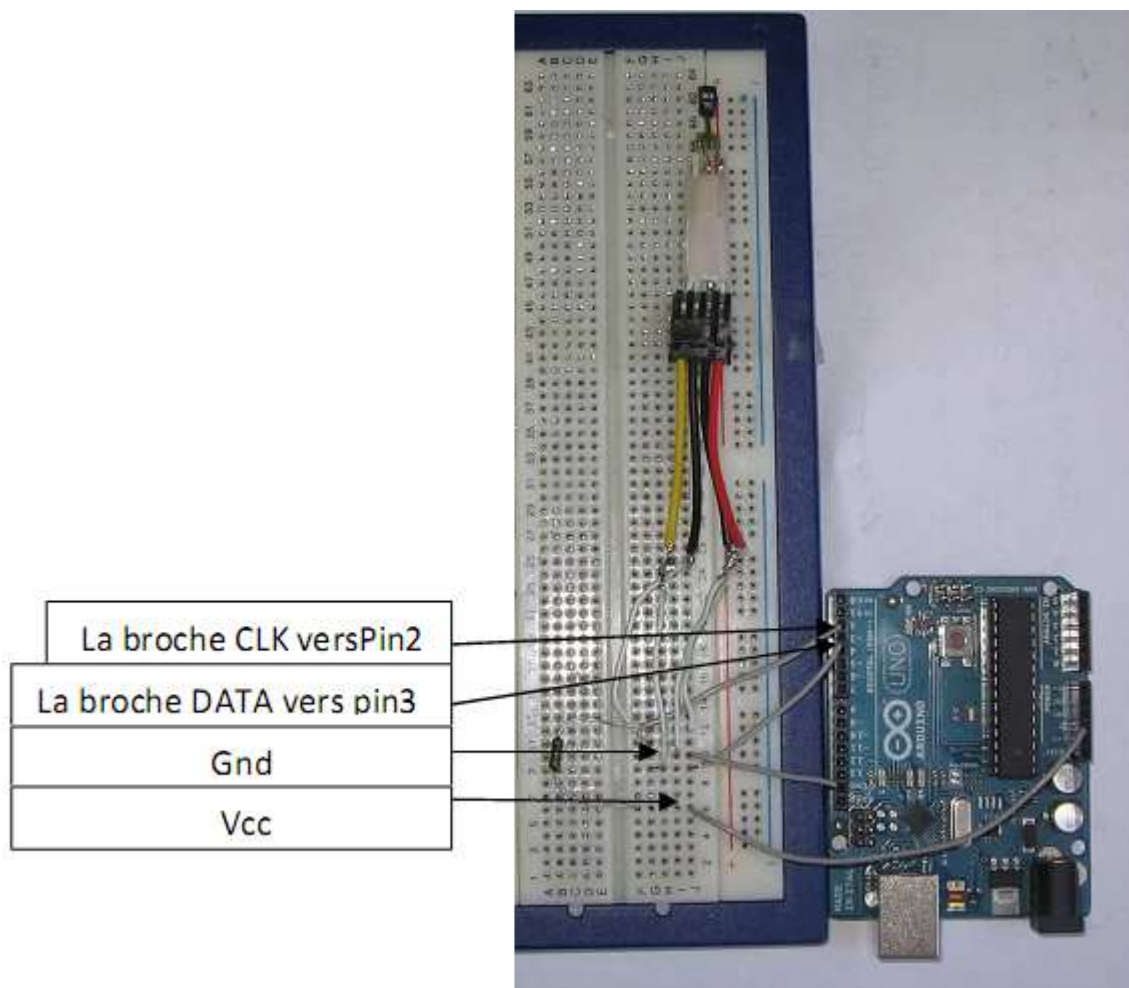


Figure 2-23 : Brochage du SHT75 Avec la carte Arduino

VI-Le module de transmission :

Il existe plusieurs technologies de transmission sans fil dans le domaine des télécommunications, et chaque une de ces technologies diffère des autres en fonction de ses performances, ses avantages, sa gamme de fréquence de travail...

Dans notre projet on a utilisé la technologie Bluetooth vu ses avantages et ses performances et sa simplicité de mise en œuvre.

VI-1-Historique :

Le mot bluetooth fait référence à un roi Viking du 10^{ème} siècle Harald Blatand.

Son nom, Blatand est devenu bluetooth dans un anglais récent. La traduction littérale de ce mot donne Harald à la dent bleue, se surnom lui viendrait du fait qu'il appréciait énormément les bleuets (plus connu sous le nom de myrtille) et qu'elles lui coloraient les dents en bleu. Il unifia la Norvège et le Danemark en préférant la consultation et la coopération plutôt que d'utiliser la puissance des armes.

L'instigateur de la norme bluetooth, Ericsson, a trouvé que ce nom serait parfait pour une technologie qui a pour but d'unifier les connections entre les ordinateurs et les appareils de télécommunication. Ericsson était un géant des télécoms norvégiens c'est pourquoi leur choix c'est porté sur un roi Viking plutôt qu'un autre.



Figure 2-23 : Symbole du Bluetooth.

VI-2 Le module bluetooth de Sparkfun :

La transmission du signal de l'ECG acquis par la carte Arduino vers l'ordinateur se fait via Bluetooth, et pour cela on a choisit le module " **BlueSMiRF** " du groupe **Sparkfun**, qui est compatible avec la carte Arduino et

sans utiliser aucun circuit externe. Ce module dispose de six broches de connexion : **Vcc, Gnd, Rx, Tx, CTS, RTS**, et deux Leds (verte et rouge), une antenne miniaturisée intégré (dans la suite on va décrire tout ces broches, et le rôle de ces deux Leds).

VI-2-1 Les caractéristiques du module BlueSMiRF :

Le module **BlueSMiRF** possède plusieurs caractéristiques techniques, ces caractéristiques sont présentées dans le tableau suivant :

caractéristiques	Propriétés
Classe	1
Consommation	25mA
Gamme de fréquence	De 2,4 à 2,524 GHz
Rapidité de modulation	De 2400 à 115200 bps
Tension d'alimentation	Entre 3,3 et 6V
Plage de température de fonctionnement	Entre -40 et +70°C
Porté	Environ 100m
Protocole de communication	UART
Dimensions	0.15x0.6x1.9 inch

Figure 2-24 : Caractéristiques du module BlueSMiRF.

VI-2-2 Brochage du module BlueSMiRF :

Comme nous l'avons vu précédemment le module BlueSMiRF dispose de six pins qui sont :

- **VCC-Gnd** : BlueSMiRF peut être alimenté avec une alimentation comprise entre 3,3V et 6V, dans notre projet nous l'avons alimenté avec les 5V que dispose la carte Arduino et ca pour éviter l'encombrement d'ajouté une autre alimentation spécialement pour ce module.

- **RX-I (Receive into)** : cette broche est utilisée pour la réception série des données, cette broche doit être connectée à la broche Tx de l'émetteur, dans notre cas elle va être connectée à la broche 1 de la carte Arduino.
- **Tx-O (Transmit from)** : comme Rx se charge de la réception des données, cette broche se charge de la transmission série des données cette broche à son tour doit être connectée à la broche Rx du récepteur, qui est la broche 2 de la carte Arduino.
- **CTS-I (Clear To Send) et RTS-O (Ready To Send)** : s'utilisent pour le contrôle du flux (dans notre projet nous les s'avons pas utilisé).

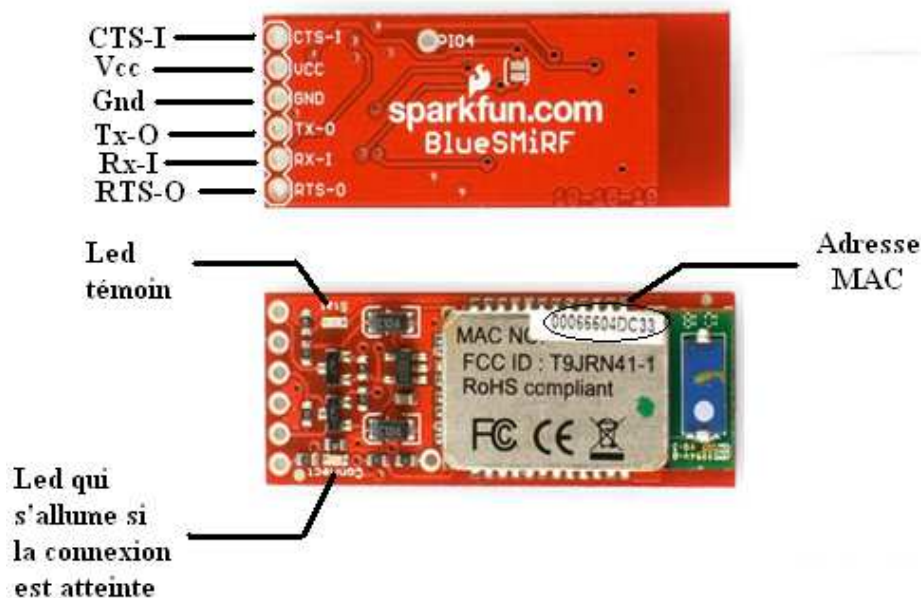


Figure 2-25 : Brochage du module BlueSMiRF.

-Brochage du module BluesMiRF avec la carte Arduino :

On utilise pour la communication avec le module Bluetooth les deux broches Tx et Rx de la carte Arduino : la pin Rx vers Tx de la carte et la pin Tx vers Rx de la carte. la figure suivante nous permet de voir ce brochage

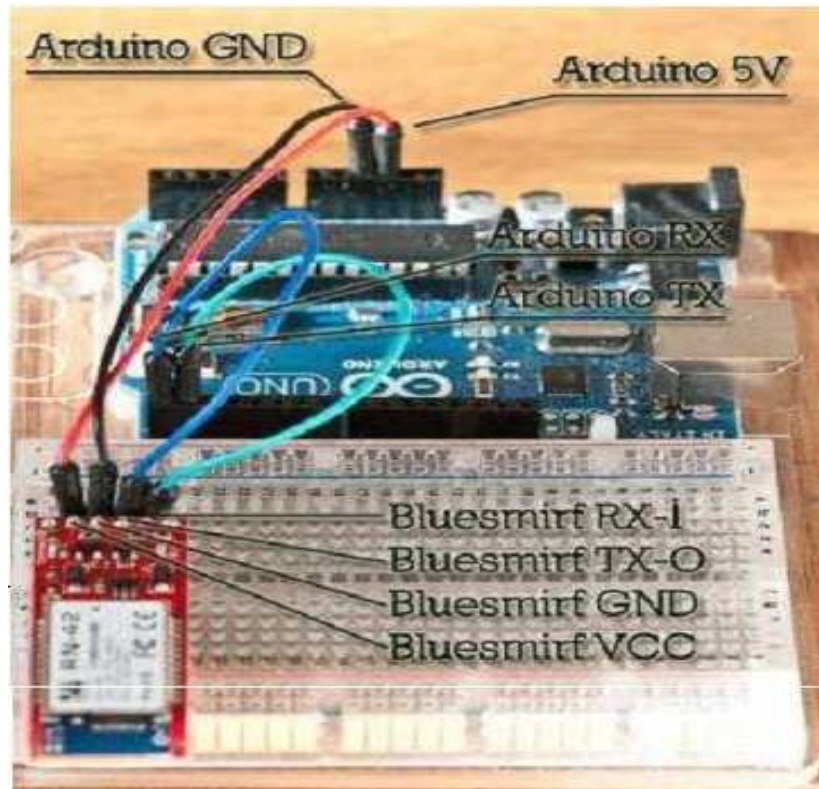


Figure 2-26 : Brochage du BlueSMiRF à la carte Arduino

Chapitre III

Programmation et conception logicielles

I- Introduction :

Ce chapitre va se consacrer sur la description des différents logiciels qu'on a utilisé au cours de notre travail, ainsi que les différentes étapes qu'on a suivi pour le développement de notre maquette.

II- 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 **Arduino EDI** (*Espace de développement Intégré*), qui porte le nom même de la carte Arduino.

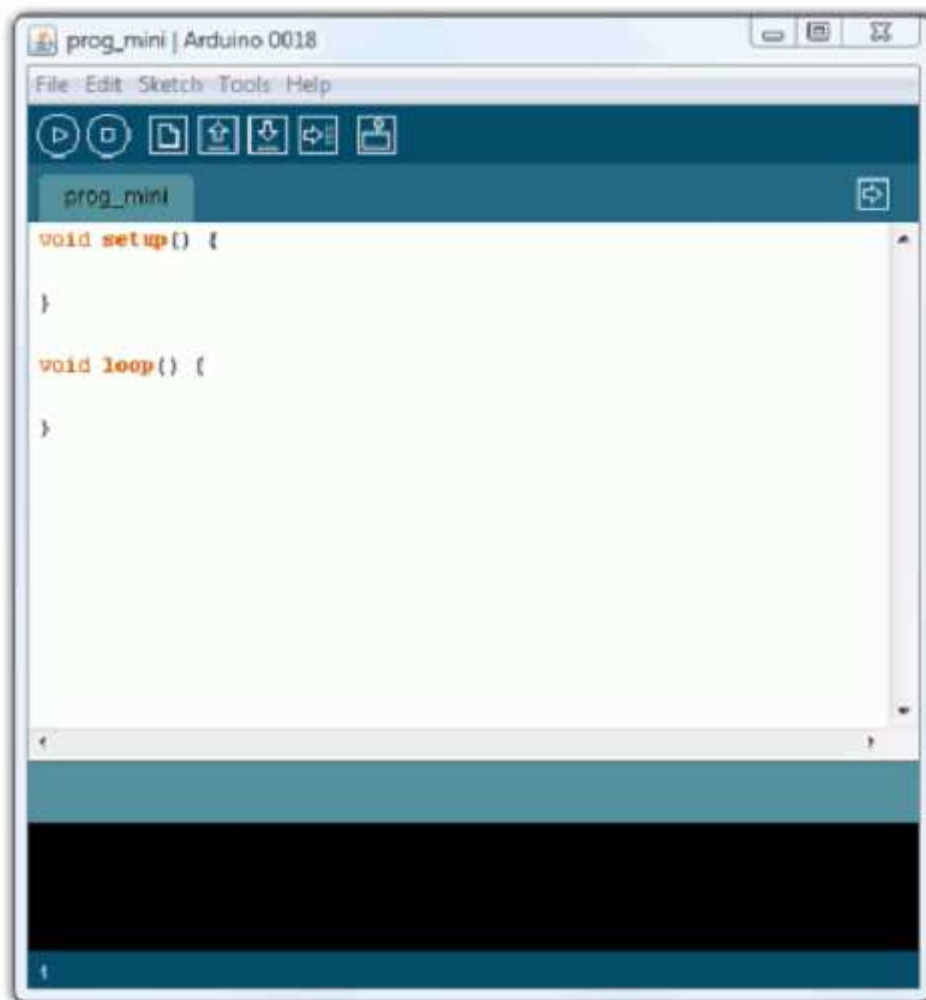


Figure 3-1 : Fenêtre principale de l'environnement de programmation ARDUINO

II-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.

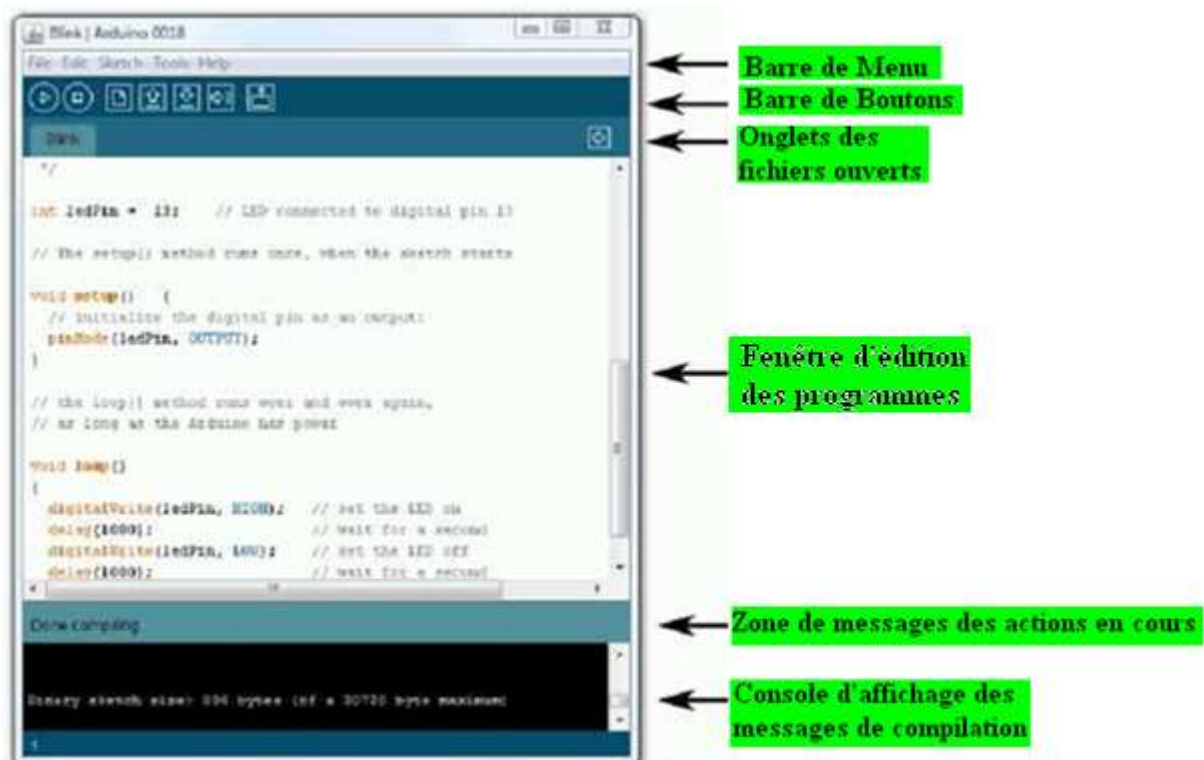


Figure 3-2 : Les différentes parties de la fenêtre principale du logiciel Arduino

Il comporte :

- ✓ une **BARRE DE MENUS** : comme pour tout logiciel une interface graphique (GUI).
- ✓ une **BARRE DE BOUTONS** : cette barre nous donne un accès direct aux fonctions essentielles du logiciel, elle importe au logiciel une simplicité d'utilisation.
- ✓ un **EDITEUR** (à coloration syntaxique) : c'est dans cette espace où on va écrire notre programme, il dispose aussi des onglets de navigation.
- ✓ une **ZONE DE MESSAGES** : permet d'afficher et d'indiquer l'état des actions en cours (ex : vérification, téléversement).

- ✓ une **CONSOLE TEXTE** : elle permet d'afficher les messages concernant le résultat de la compilation du programme (il nous indique s'il y a des erreurs).
- ✓ un **MONITEUR SERIE** : ce moniteur est utilisé pour la visualisation ou les données (message, valeurs, caractères) transmises vers l'ordinateur via le câble USB connecté à la carte Arduino et d'envoyer des caractères vers la carte Arduino.

Cette fonctionnalité permet une mise au point de la facilité des programmes et d'afficher sur l'ordinateur l'état des variables d'un programme où il y a des calculs ou des mesures, aussi le résultat de la conversions analogique-numérique : un élément essentiel pour améliorer, tester et corriger ses programmes.

Nous l'avons utilisé dans notre projet pour tester si la carte fait la conversion analogique/numérique pour le signal de l'ECG, aussi pour vérifier si la carte transmits ces valeurs convertis, la valeur de la température et la valeur de l'humidité.



Figure 3-3 : Fenêtre associée au moniteur série

II-1-1-Description de la barre des boutons :

La barre des boutons nous permet de vérifier la syntaxe, le transfère 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 bouton ou icônes :



Figure 3-4 : 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.



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



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 ca peut se faire en cliquant sur cette icône.



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

II-1-2-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 3-5 : 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 utilisé ainsi que la fonction de chargement du bootloader dans l'ATmega.
- **Help (Aide) :** ce menu est fait pour donner de l'aide concernant les différents problèmes rencontrés au niveau du logiciel Arduino.

III- 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 sur 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 en dessus de l'onglet d'édition des programmes indiquant qu'elle est l'erreur et la ligne où se trouve (pour la correction des erreurs il suffit de consulter l'aide disponible dans la barre des menus). 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, il suffit juste de la mettre sous tension et de lui brancher les composants ou les éléments nécessaire pour le bon déroulement du processus à exécuter.

IV- Le logiciel BlueSoleil :

La transmission de notre signal de l'ECG et les valeurs de la température et de l'humidité vers l'ordinateur se fait via bluetooth, mais il se trouve que les bluetooths des ordinateurs ne transmettent ou reçoivent que des objets ou fichiers (images, texte, vidéo...), or dans notre cas on doit transmettre un ensemble de données et non pas des objets ou fichiers et de manière continue.

Pour la résolution de ce problème on doit créer un **PAN** (*Personal Area Network*). Il existe un logiciel qui nous permet de créer ce type de réseau avec différents éléments (PDA (*Personal Digital Assistant*), ordinateurs, imprimantes...), et la transmission de données va se faire de manière continue. Ce logiciel est **BlueSoleil**.

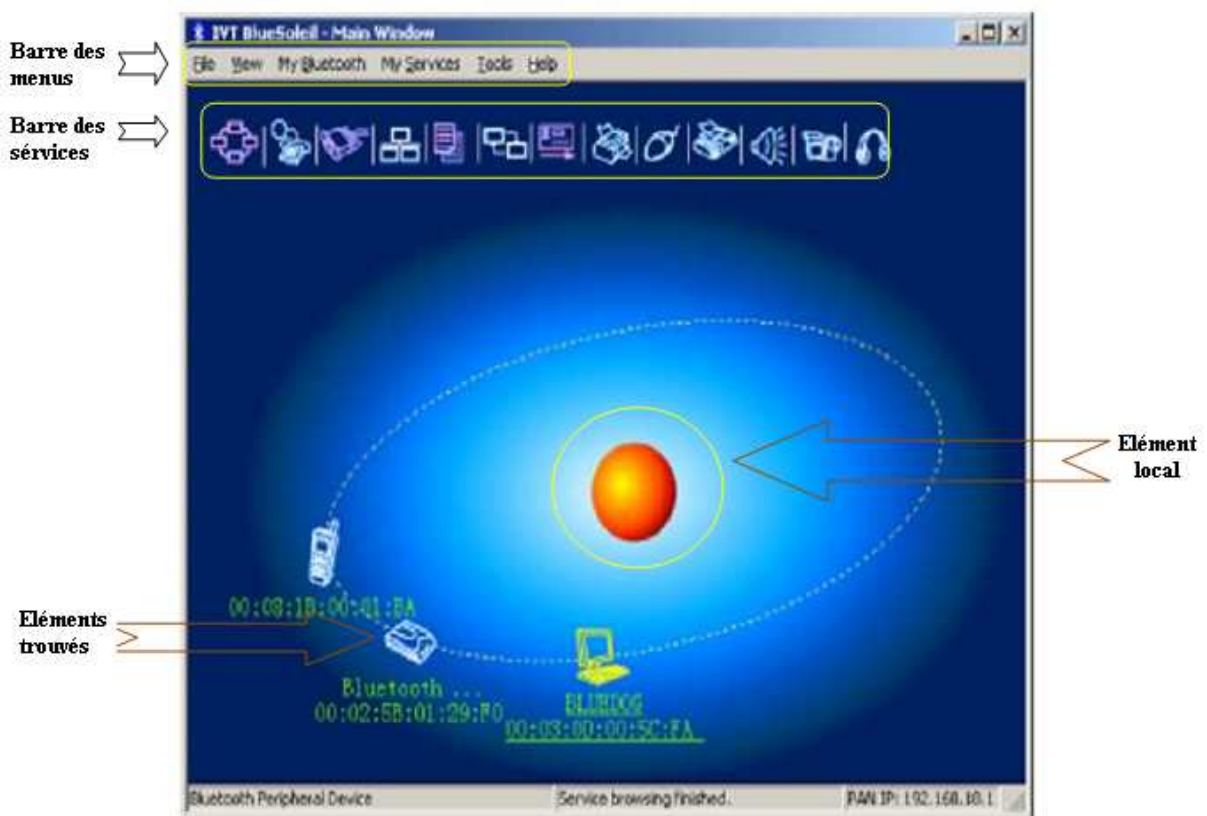


Figure 3-6 : Fenêtre principale de Bluesoleil

IV-1 Description de BlueSoleil :

Bluesoleil comporte un nombre important de fonction et de services qui sont :

- ✓ Possibilité de créer un PAN.
- ✓ Création d'un port série virtuel .
- ✓ Accès au réseau LAN (*Local Area Network*).
- ✓ Transfert de fichiers.
- ✓ Transfert d'images.
- ✓ Partage d'imprimante et Fax.

Ce qui nous intéresse dans notre projet est la création du PAN, aussi la sécurisation de la transmission des données du moment BlueSoleil permet une sécurisation en trois niveaux (basse, moyenne, et haute).

Lors du démarrage du Bluetooth, BlueSoleil fait une recherche de dispositifs ayant un Bluetooth allumé, et pour se connecter à un dispositif il suffit de faire un double clique dessus et l'envoi de données commence, mais parfois le dispositif comme dans notre cas demande une clé pour accepter la communication. Le module BlueSMiRF demande une clé dit d'appairage qui est par défaut 1234.

V- Le logiciel de programmation graphique LabVIEW :

Après la création du PAN avec le logiciel BlueSoleil, les données sont transmises de la carte Arduino vers l'ordinateur via bluetooth, mais il se trouve qu'on a besoin d'une interface graphique pour visualiser ces données, pour ce faire on a choisi le logiciel LabVIEW (*Laboratory Virtual Instrumentation Engineering Workbench*), qui est un logiciel de développement d'applications, répondant à la même finalité que les nombreux systèmes de développement en langage C ou Basic. Cependant, LabVIEW se distingue des autres logiciels sur au moins un point important.

En effet, la majorité d'entre eux utilisent des langages à base de texte dont la programmation consiste à empiler des lignes de code alors que LabVIEW utilise un langage de programmation graphique (le langage G) pour créer un programme sous forme de diagramme.

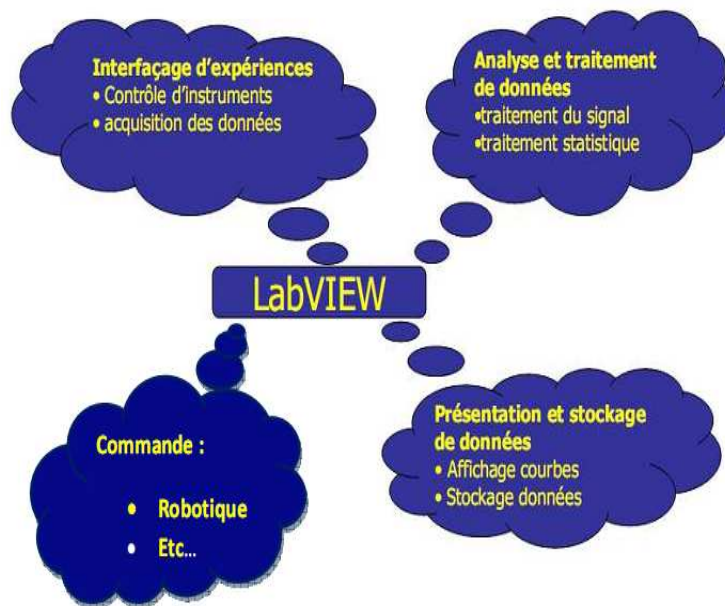


Figure 3-7 : Les différentes fonctions de LabVIEW

LabVIEW offre des bibliothèques étendues de fonctions et de routines répondant à la plupart des besoins en programmation. Pour les systèmes d'exploitation, LabVIEW contient également des bibliothèques de fonctions spécifiques à l'acquisition de données provenant des instruments connectés sur une simple liaison série. Il existe aussi des bibliothèques dédiées à la présentation et à l'analyse des données et à leur stockage. LabVIEW intègre un jeu complet d'outils de mise au point qui permettent de placer des points d'arrêt,

d'animer l'exécution du programme en mettant en évidence le cheminement des données et d'exécuter pas à pas le programme.

V-1-Les instruments virtuels :

Les programmes LabVIEW s'appellent des Instruments Virtuels (VIs). Ces VIs ont deux parties principales : la Face Avant (Front Panel), le Diagramme (Block Diagram) :

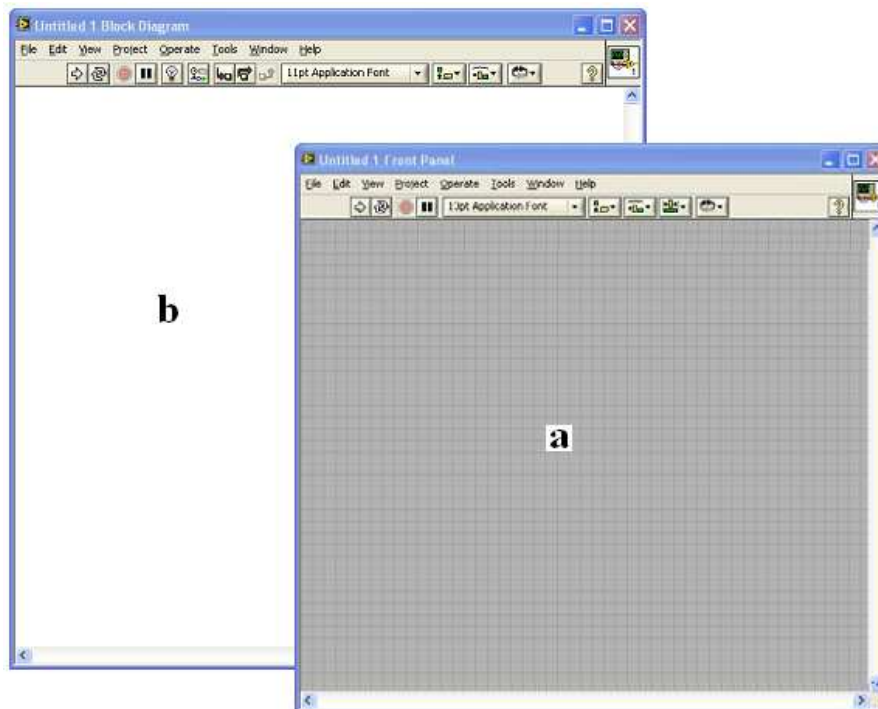


Figure 3-8 : Les deux fenêtres principales de LabVIEW : (a) Face avant, (b) Diagramme

❖ La face avant :

La face-avant est l'interface utilisateur du VI. Elle est construite à l'aide de commandes et d'indicateurs, qui sont respectivement les terminaux d'entrées et de sorties interactifs du VI. Les commandes sont des boutons rotatifs, des boutons-poussoirs, des cadrans et autres périphériques d'entrée. Les indicateurs sont des graphes, des LED et autres afficheurs. Les commandes simulent les périphériques d'entrée d'instruments et fournissent les données au diagramme du VI. Les indicateurs simulent les périphériques de sortie d'instruments et affiche les données que le diagramme acquies ou génère.

Les différents indicateurs, et outils de décoration de la face avant sont arrangés dans une palette appelée **palette de commande** (Figure 3-9) :

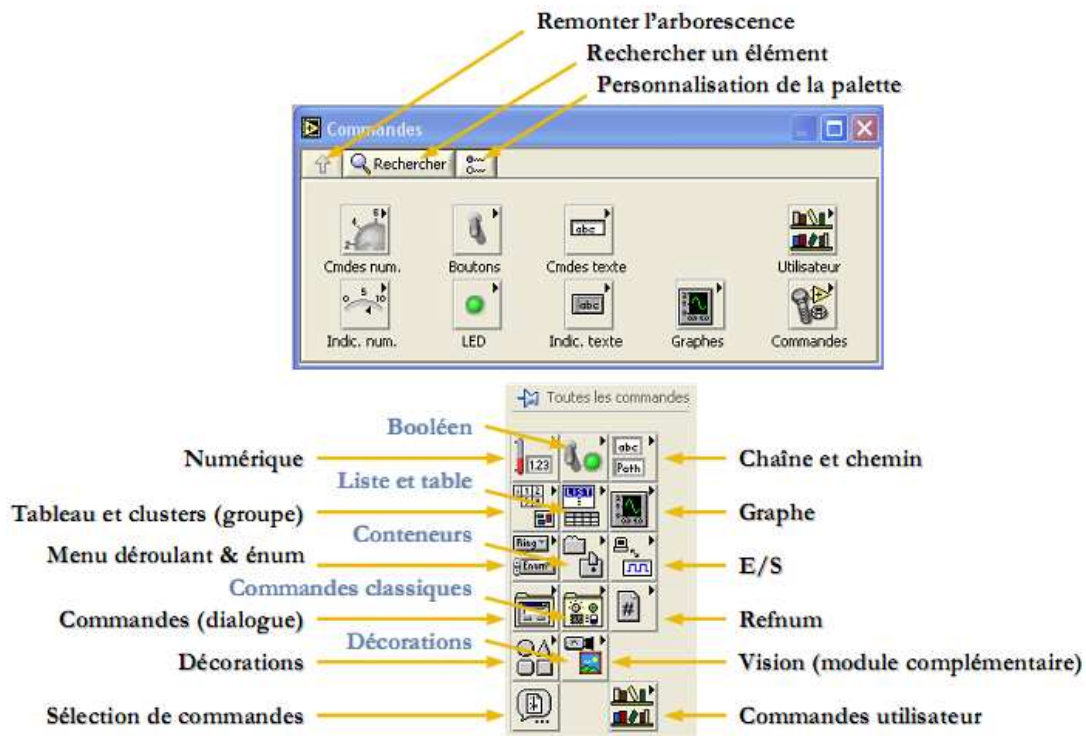


Figure 3-9 : Palette de commande

❖ Le diagramme :

Après avoir construit la face-avant, le code doit être ajouté en utilisant les représentations graphiques des fonctions pour contrôler les objets de la face-avant. Le diagramme contient ce code source graphique. Les objets de la face-avant apparaissent comme des terminaux sur le diagramme. On ne peut pas supprimer un terminal du diagramme. Le terminal disparaît uniquement après que son objet correspondant dans la face-avant a été supprimé.

Les objets du diagramme (portes logiques outils d'arithmétique, les différentes boucles...) sont arrangés dans une palette appelée **palette de fonctions** (Figure 3-10) :

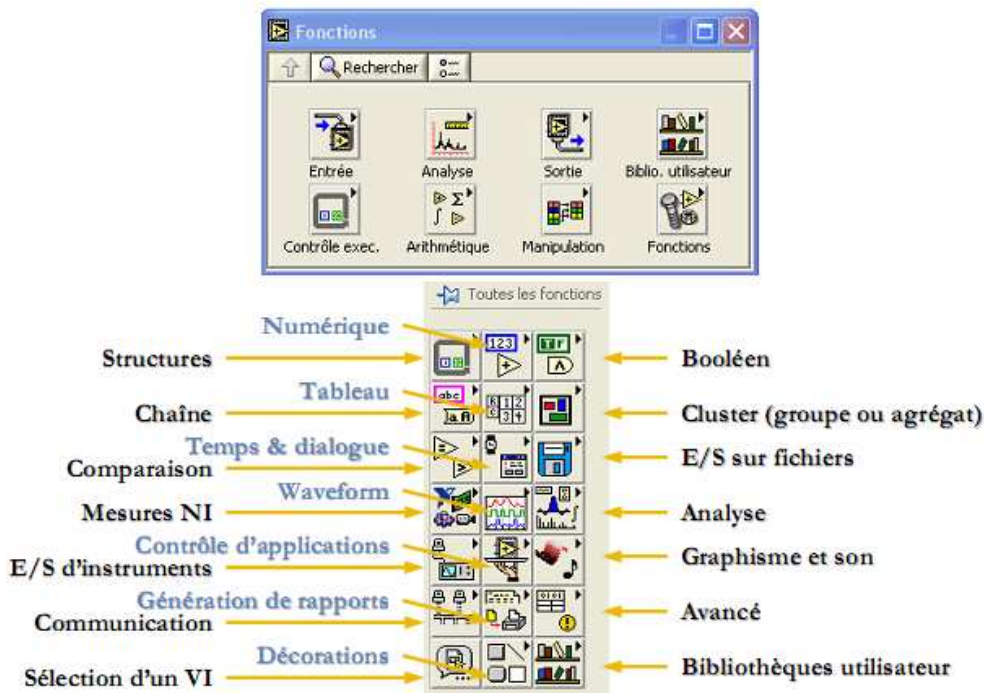


Figure 3-10 : Palette de fonctions.

Il existe une troisième palette qui peut être utilisée par les deux faces, c'est la palette d'outils (Figure 3-11) :

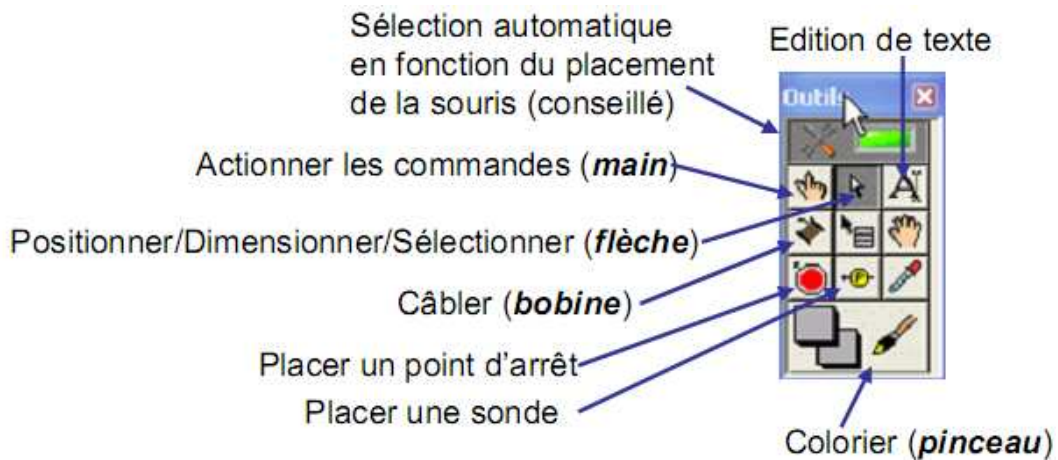


Figure 3-11 : Palette d'outils

V-2 Les types de données de base utilisés dans LabVIEW :

Dans LabVIEW toute donnée a son type et à chaque type de base correspond une couleur (Figure 3-12). Ainsi, par exemple les types numériques discrets sont en bleus (entiers signés, non signés, sur 8, 16, 32 ou 64 bits). Voici un tableau qui illustre tout ces types de données et leur symbole associé :

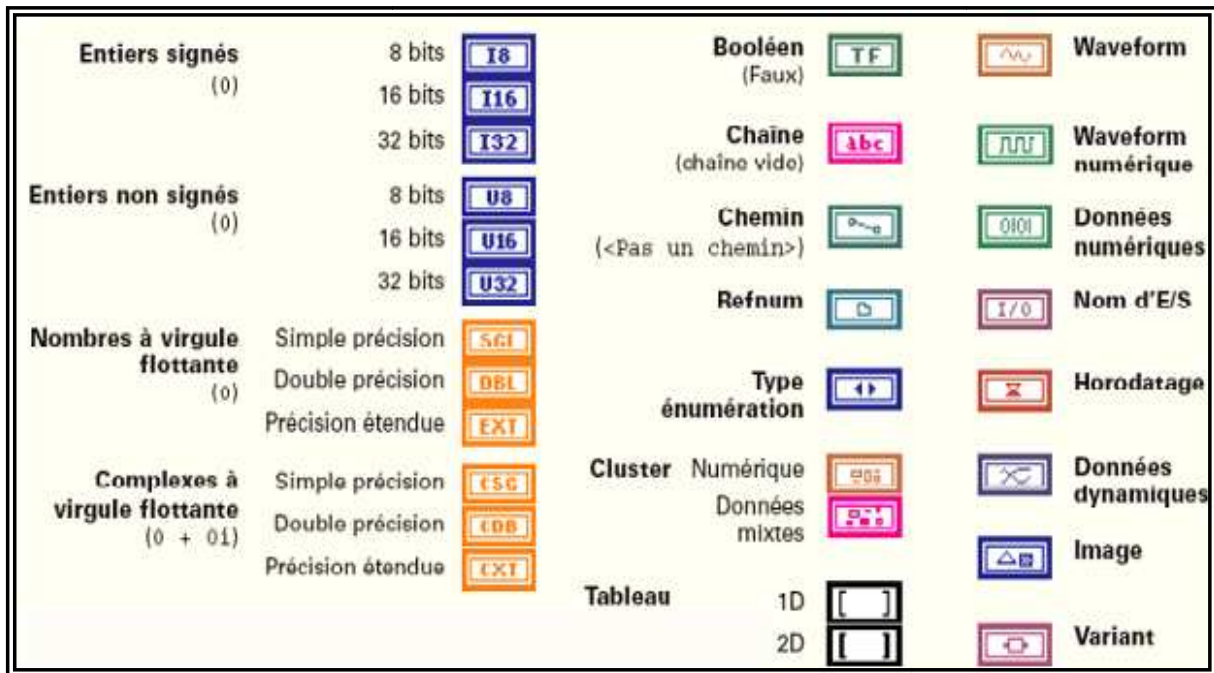


Figure 3-12 : Les différents types de données utilisés dans l'environnement LabVIEW

V-3- Description de notre interface réalisée :

Afin de visualiser toutes les données transmises par la carte Arduino, on a réalisé une interface contenant un écran genre d'un oscilloscope destiné pour la visualisation du signal de l'ECG, et petite fenêtre pour afficher la valeur de la température et la valeur de l'humidité, pour cela on a utilisé des outils pour l'acquisition des données et des outils pour les visualiser.

L'acquisition des données va se faire en utilisant une bibliothèque offerte par le logiciel LabVIEW qui nous permet de communiquer avec des instruments reliés au pc. Cette bibliothèque est appelée VISA pour (*Virtual Instrumentation Software Architecture*) *Architecture Virtuelle du Logiciel d'Instrumentation*, elle permet de configurer tout type de communication comme étant une communication série

Bluetooth comme étant un port série qui comporte plusieurs paramètres (débit en baud, nombre des bits de parité, le bit d'arrêt, contrôle du flux...)

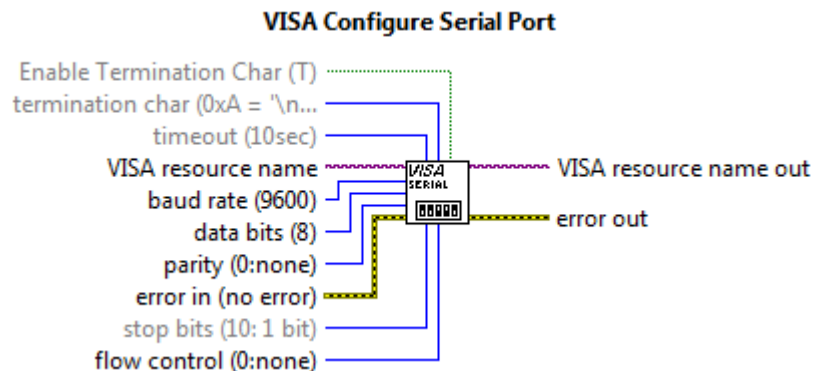


Figure 3-15 : La fonction " VISA Configure Serial Port "

V-3-2- Configuration de la taille du tampon d'entrée/sortie :

La palette de VISA contient aussi une fonction qui nous permet de définir la taille du tampon d'entée/sortie (c.à.d. pour la transmission et la réception des données), cette fonction est donnée sous le nom **VISA Set I/O Buffer Size Fonction**, elle peut définir la taille du tampon pour la réception uniquement (le masque est de 16 bits), ou pour la transmission uniquement (le masque est de 32 bits), ou bien pour la transmission et la réception au même temps (dans ce cas le masque est de 48 bits).

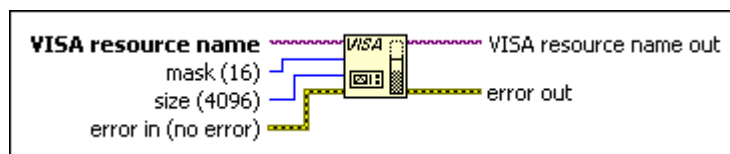


Figure 3-16 : la fonction "VISA Set I/O Buffer Size Fonction"

La fonction qui reste pour lire les données provenant du port, c'est la fonction **VISA Read Fonction**, cette fonction nous permet de lire les données sous forme d'octets et de les transférer vers l'instrument de visualisation ou de sauvegarde.

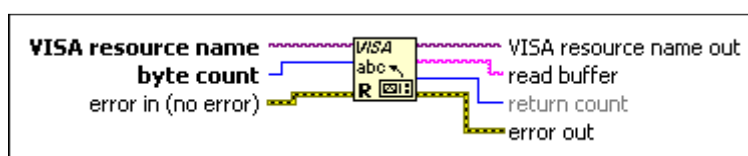


Figure 3-17 : La fonction "VISA Read Fonction "

Toutes ces fonctions que nous avons citées vont être placées dans la face diagramme de LabVIEW, à ce niveau il nous reste qu'à définir les éléments de la face avant qui sont tout simplement un indicateur de type **graphe déroulant**, et une petite fenêtre pour la visualisation des valeurs de la température et la valeur de l'humidité

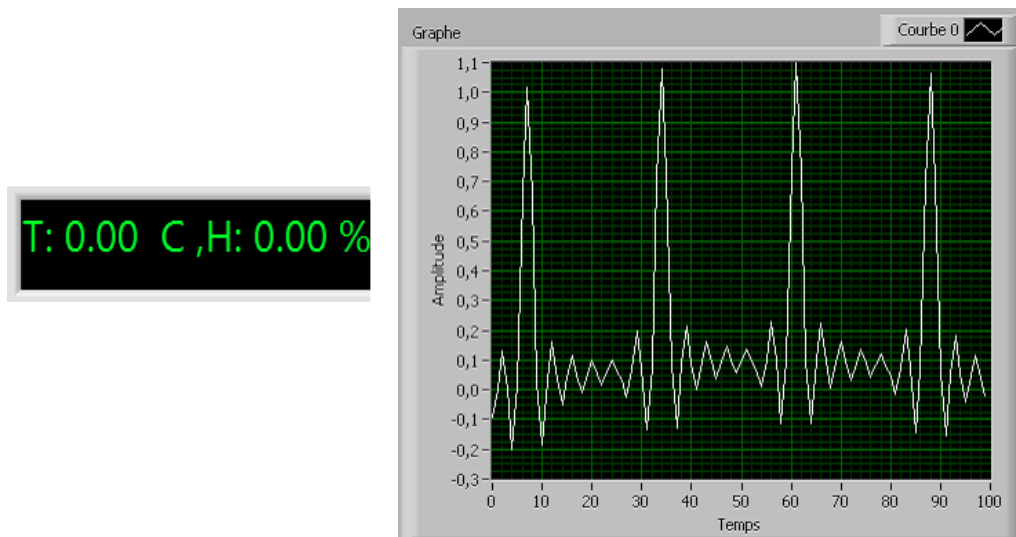


Figure 3-17 : L'afficheur pour la température et l'humidité, et l'indicateur « Graphe déroulant »

Après la conception des deux faces on a obtenu l'interface suivante :

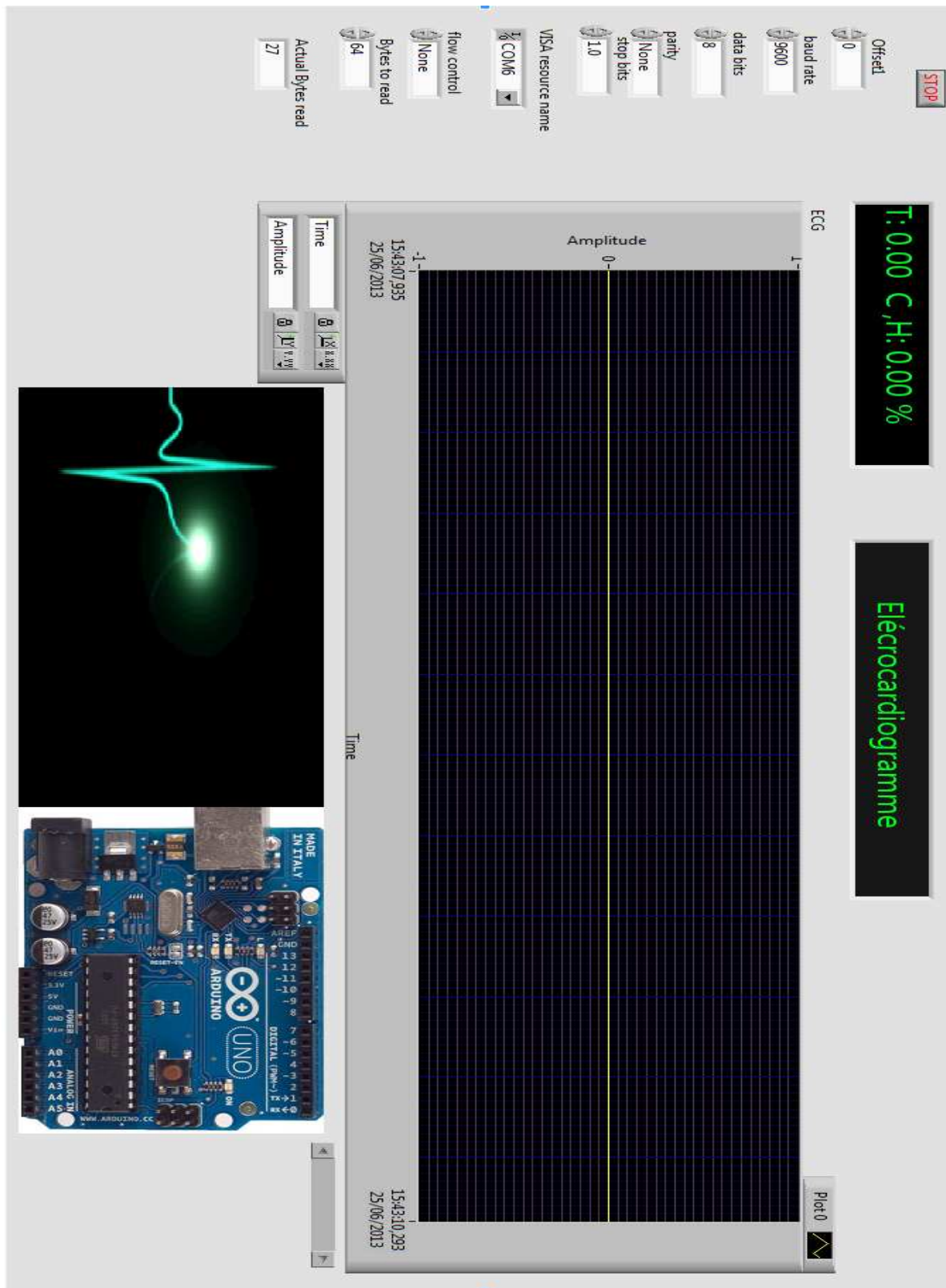
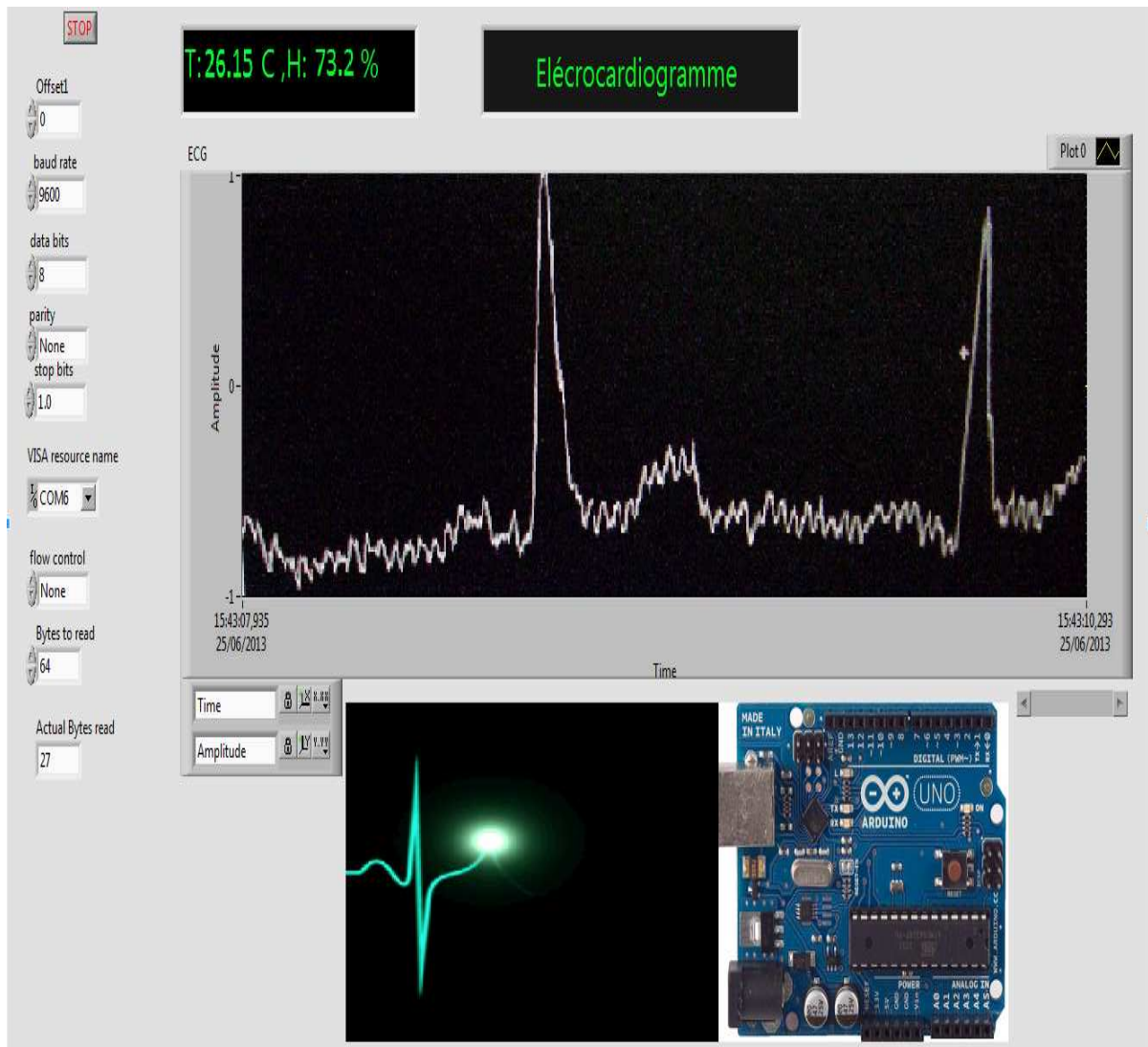


Figure 3-18 : La face avant

VI-Résultat :

Après avoir réalisé notre interface, on a procédé à la visualisation des résultats obtenus d'un teste fait au niveau du laboratoire projet. On a pu faire la transmission du signal ECG et les valeurs de la température et de l'humidité de la carte Arduino vers un pc ensuite les visualiser sous LabVIEW mais on a pas pu trouver l'influence de la température et de l'humidité sur le signal ECG avec les testes qu'on a fait car il n'y avait pas de variations importantes de la température et de l'humidité. Il fallait se déplacer vers des milieux où il y a des variations remarquables de la température et de l'humidité, mais puisque il y avait pas de temps suffisant pour le faire on s'est limité sur les résultats obtenus du teste cité précédemment (**Figure 3-20**).

**Figure3-20 : Visualisation des résultats sous LABVIEW**

Conclusion

Conclusion

Dans notre travail mené tout au long de ce mémoire, nous avons conçu et réalisé un système de télésurveillance médicale à base d'une carte ARDUINO avec une transmission des données via bluetooth vers un ordinateur où la visualisation des données a été faite en utilisant le logiciel de programmation graphique LabVIEW.

Le premier chapitre de cette thèse a été consacré sur les principes de base du signal ECG, et des généralités sur le système cardiovasculaire, ainsi qu'un aperçu sur l'électrocardiogramme et les différentes dérivations permettant de le recueillir en appuyant sur le théorème d'Einthoven et une définition du Holter ECG.

Dans le second chapitre nous avons fait une description détaillée des différents étages constituant notre chaîne d'acquisition, partant des électrodes en passant par la partie analogique constituée principalement d'un amplificateur d'instrumentation et d'un circuit de filtrage, et enfin la numérisation du signal acquis en utilisant une carte ARDUINO, qui nous a permis aussi de mesurer la température et l'humidité avec en utilisant le capteur numérique SHT75.

Après avoir acquis le signal, une transmission vers un ordinateur de ces données acquises est assurée par un module bluetooth pour mettre en évidence la mobilité du patient, ainsi pour faire une visualisation du signal ECG avec des outils beaucoup plus performants.

La dernière partie de cette thèse est consacrée pour décrire les différents logiciels utilisés au cours de ce travail, et la réalisation d'une interface de visualisation sous LabVIEW qui nous a permis de visualiser le signal ECG ainsi que les valeurs de la température et celles de l'humidité pour une étude des effets de la variation de ces deux grandeurs sur le signal ECG.

Enfin, nous avons eu la chance de rencontrer et d'apprendre plusieurs techniques tel que les logiciels de programmation et de simulation ainsi que l'approfondissement des connaissances dans le domaine de l'acquisition temps-réel.

Nous souhaitons que ce modeste travail servira d'avantage pour les promotions qui viennent et de leur donner une image réelle sur l'importance de la partie pratique de l'électronique.

Bibliographie

Bibliographie

[1] Jean Send, "**Guide Pratique ECG**", p :14-15, 22-27.

[2] El Mimouni El Hassan et Karim Mohammed, "**Design and Implementation of an Embedded System for Ambulatory Cardiac Monitoring**", Journal of Telecommunications and Information Technology, 1/2012.

[3] Leif Sörnmo et Pablo Laguna, "**Bioelectrical Signal Processing in Cardiac and Neurological Application**", p: 24, 426-427.

Norsheila Fisal, Rozeha Abd Rashid, Mohd Adib Sarijari, Haslinah Mohd Nasir "ECG Monitoring System Using Wireless Sensor Network (WSN) for Home Care Environment"

[4] Anh Dinh "**Heart Activity Monitoring on Smartphone**", 2011 International Conference on Biomedical Engineering and Technology IPCBEE vol.11 (2011) IACSIT Press, Singapore.

[5] Gurdhyan Singh, Harish Chandra et Soami P. Satsangee, "**FABRICATION OF USB BASED ELECTROCARDIOGRAM MACHINE**", XXXII NATIONAL SYSTEMS CONFERENCE, NSC 2008, December 17-19, 2008.

[6] Mehmet ENGİN¹, Tayfun DALBASTI, Saygın BILDIK¹, Turan KARIPÇIN¹, Erkan Zeki ENGİN, "**A DESIGN OF PORTABLE HEART-RATE MONITORING SYSTEM**", M.ENGİN et al. / IU-JEEE Vol. 10(2), (2010), 1201-1205.

[7] J.J.Segura-Jucirez, D. Cuesta-Frau and L Samblas-Pena, "**A MICROCONTROLLER-BASED PORTABLE ELECTROCARDIOGRAPH**", SYSTEM, 0-7803-8163- 0 2003 IEEE.

[8] Ashraf A Tahat, "**Body Temperature and Electrocardiogram Monitoring Using an SMS-Based Telemedicine System**", 978-1-4244-2966-0/09, IEEE 2009.

[9] Md. Kafiul Islam, Md. Asaduzzaman, Tanvir Ahammad, Tasnuva Faruk Nowrin, "**Embedded Programmable Web-based ECG Monitoring & Detection System Using a Fast Algorithm**", 978-1-4244-6775-4/10, 2010 IEEE.

[10] Brian A. Walker, Ahsan H. Khandoker, and Jim "**Black Low Cost ECG Monitor for Developing Countries**", 978-1-4244-3518-0/09, 2009 IEEE.

- [11] Gavin P. Shorten and Martin J. Burke, "**A precision ECG signal generator providing full Lead II QRS amplitude variability and an accurate timing profile**", 31st Annual International Conference of the IEEE EMBS.
- [12] Shi Zhang, Jinshuan Zhao, Lihuang She, and Guohua WangMinneapolis, "**A Novel Pocket Intelligent One Lead ECG Monitor Based on Fingers Touching**", 978-1-4244-4131-0/09, 2009 IEEE.
- [13] Min Zeng, Il-Yong Chung, and Jeong-A Lee et Jeong-Gun Lee, "**An On-Node Intelligence based Energy Efficient ECG Monitoring System**", 978-1-4577-1268-5/11,2011 IEEE.
- [14] Tom Torfs, Refet Firat Yazicioglu, Sunyoung Kim, Dilpreet Buxi, Inaki Romero, Jacqueline Wijsman, "**Ultra Low Power Wireless ECG system with Beat Detection and Real Time Impedance Measurement**", 978-1-4244-7270-3/10, 2010 IEEE.
- [15] Dr.(Mrs).R.Sukanesh, S.Palanivel Rajan, S.Vijayprasath, N.S.Aishwarya, P.Gracy Angela, "**Intelligent Wireless Mobile Patient Monitoring System**", 978-1-4244-7770-8/10,2010 IEEE.
- [16] Mohamed Fezari, Mounir Bousbia-Salah, and Mouldi Bedda, "**Microcontroller Based Heart Rate Monitor**", The International Arab Journal of Information Technology, Vol. 5, No. 4, October 2008.
- [17] Majid A. Al-Tae, Nadine A. Jaradat, Dima M. Abu Ali , "**Mobile Phone-Based Health Data Acquisition System Using Bluetooth Technology**", Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT), IEEE 2011.
- [18] Liang Kai, Zhang Xu, Wang Yuan, Huang Suibiao, Guan Ning, Pang Wangyong, Li Bin, Chen Hongda , "**A System of Portable ECG Monitoring Based on Bluetooth Mobile Phone**", 978-1- 61284-704-7/11, 2011IEEE.
- [19] Noureddine BELGACEM , Fethi BEREKSI-REGUIG, "**Bluetooth Portable Device for ECG and Patient Motion Monitoring**", « Nature & Technology » Review. Issue n° 04/January 2011. Pages 19 to 23.
- [20] Ying-Chien Wei, Yu-Hao Lee, and Ming-Shing Young, "**A Portable ECG Signal Monitor and Analyzer**", 978-1-4244-1748-3/08, 2008 IEEE.
- [21] Dr.Ali Moti Nasrabadi, Minoo Heidari Kani, "**Design of ECG acquisition and transmission via Bluetooth with heart disease diagnosis**",978-1-4244-9338-8/11, IEEE, 2011.
- [22] Piroon Kaewfoongrungsie, Nipon Theera-Umpon, and Sansanee Auephanwiriyaikul, "**ECG HOLTER RECORDER VIA MOBILE PHONE**", The 3rd International Symposium on Biomedical Engineering (ISBME 2008).

[23] Nouredine BELGACEM, Said ASSOUS, Fethi BEREKSI-REGUIG, "**BLUETOOTH PORTABLE DEVICE AND MATLAB-BASED GUI FOR ECG SIGNAL ACQUISITION AND ANALISYS**", 7th International Workshop on Systems, Signal Processing and their Applications (WOSSPA).

[24] Daniel Lucani, , Giancarlos Cataldo, , Julio Cruz, Guillermo Villegas, and Sara Wong, "**A portable ECG monitoring device with Bluetooth and Holter capabilities for telemedicine applications**", Proceedings of the 28th IEEE EMBS Annual International Conference New York City, USA, Aug 30-Sept 3, 2006.

[25] Borromeo S, Rodriguez-Sanchez C, Machado F, Hernandez-Tamames JA, de la Prieta R, "**A Reconfigurable, Wearable, Wireless ECG System**", Proceedings of the 29th Annual International Conference of the IEEE EMB Cité Internationale, Lyon, France August 23-26, 2007.

Sites Web :

[1] www.MED-FES.COM.

[2] www.arduino.cc.

[3] www.abcelectronique.com.

[4] www.fabricsincer.com.

[5] www.ni.com.

[6] www.alldatasheet.com.

[7] www.bibliodunet.com.

[8] www.texas-instrument.com.

Annexe

1 Sensor Performance Specifications⁽¹⁾

Parameter	Conditions	Min.	Typ.	Max.	Units
Humidity					
Resolution		0.5	0.03	0.03	% RH
		8	12	12	bit
Repeatability			±0.1		% RH
Accuracy ⁽²⁾ & Interchangeability		see figure 1			
Nonlinearity	10 - 90 %RH	<1 ⁽⁵⁾		±3 ⁽⁵⁾	% RH
Range		0		100	% RH
Response time	1/e (63%) slowly moving air		4		s
Hysteresis			±1		% RH
Long term stability	Typical		< 1		% RH/yr
Temperature					
Resolution		0.04	0.01	0.01	°C
		0.07	0.02	0.02	°F
		12	14	14	bit
Repeatability			±0.1		°C
			±0.2		°F
Accuracy		see figure 1			
Range		-40		123.8	°C
		-40		254.9	°F
Response Time	1/e (63%)	5		30	s

Table 1 Sensor Performance Specifications

1.1 Converting the digital output to physical values

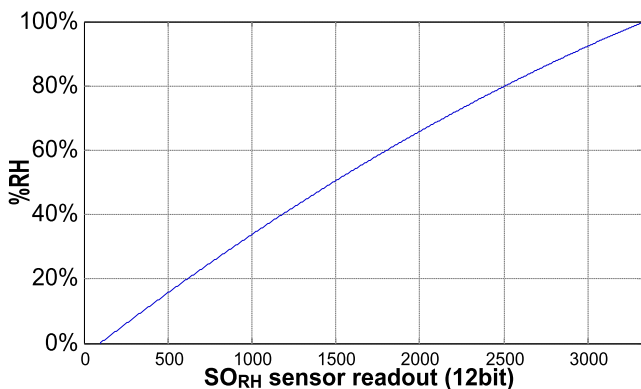
1.1.1 Humidity

To compensate for the non-linearity of the humidity sensor and to obtain the full accuracy it is recommended to convert the readout with the following formula:

$$RH_{linear} = c_1 + c_2 \cdot SO_{RH} + c_3 \cdot SO_{RH}^2$$

$c_1 = -4$ $c_2 = 0.0405$ $c_3 = -2.8 \cdot 10^{-6}$ for 12bit SO_{RH}
 $c_1 = -4$ $c_2 = 0.648$ $c_3 = -7.2 \cdot 10^{-4}$ for 8bit SO_{RH}

For simplified, less computation intense conversion formulas see application note "RH Non-Linearity Compensation".



⁽¹⁾ For operation within normal operation range as described in Chapter 3, RH accuracy at 25°C

⁽²⁾ Not including non-linearity

⁽³⁾ The default measurement resolution of 14bit (temperature) and 12bit (humidity) can be reduced to 12 and 8 bit through the status register.

⁽⁴⁾ Where SO_{RH} is the sensor output for relative humidity

⁽⁵⁾ Min. value after compensation with formula in chapter 1.1.1, Max. value without any compensation

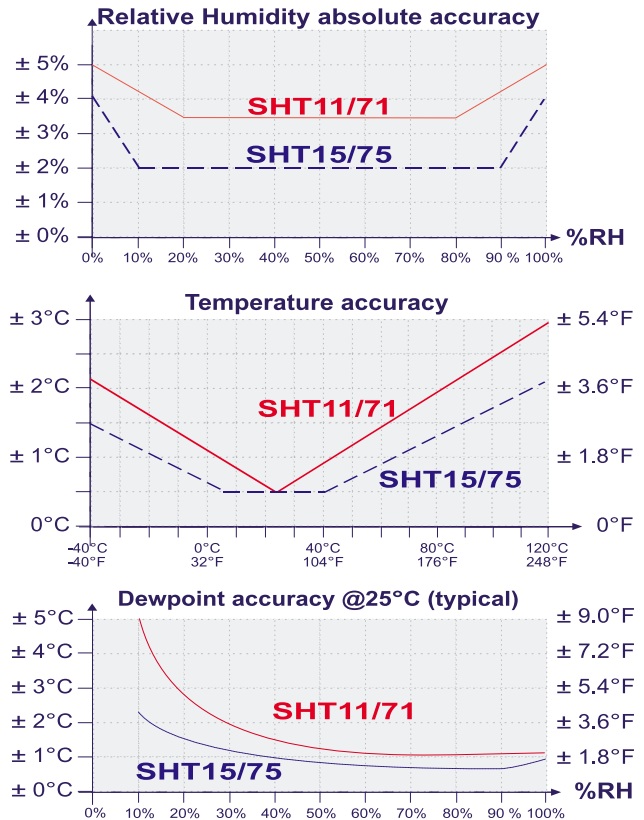


Figure 1 Rel. Humidity, Temperature and Dewpoint accuracies

For temperatures significantly different from 25°C (~77°F) the temperature coefficient of the RH sensor should be considered:

$$RH_{true} = (T_c - 25) \cdot (t_1 + t_2 \cdot SO_{RH}) + RH_{linear}$$

with $t_1 = 0.01$; $t_2 = 0.00008$ for 12bit SO_{RH} ; $t_2 = 0.00128$ for 8bit SO_{RH}
 This equals ~0.12%RH / °C @ 50%RH

1.1.2 Temperature

The temperature sensor is very linear by design. Use the following formula to convert from digital readout to temperature: Temperature = $d_1 + d_2 \cdot SO_T$

Use the appropriate table entries for 5V or 3V.

SO_T	Celsius		Fahrenheit	
	d_1	d_2	d_1	d_2
14bit 5V	-40	0.01	-40	0.018
12bit 5V	-40	0.04	-40	0.072
14bit 3V	-38.4	0.0098	-37.1	0.0176
12bit 3V	-38.4	0.0392	-37.1	0.0704

This equals a voltage dependency of ~ -0.2°C/V @ 25°C

1.1.3 Dewpoint

See application note "Dewpoint calculation" for more information.

2 Serial Interface

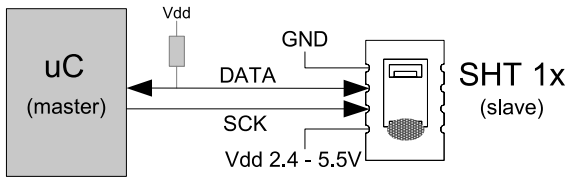


Figure 2 Typical application circuit

2.1 Power Pins

The device requires a voltage supply between 2.4V and 5.5V. After powerup the device requires 11ms to reach its "sleep" state. No commands should be sent before that time. Power supply pins (VDD, GND) may be decoupled with a 100 nF capacitor.

2.2 I/O Pins (Bidirectional 2-wire Interface)

See Table 5 for a detailed IO characteristics.

2.2.1 Serial clock input (SCK)

The SCK is used to synchronize the communication between a master and the SHT1x/SHT7x. Since the device contains fully static logic there is no minimum SCK frequency.

2.2.2 Serial data (DATA)

The DATA tristate pin is used to transfer data in and out of the device. DATA **changes at the falling edge** and is **valid on the rising edge** of the serial clock SCK. An external pull-up resistor is required to pull the signal high. (See Figure 2). Pull-up resistors are often included in I/O circuits of microcontrollers.

2.2.3 Command sequence

To initiate a transmission a "Transmission Start" sequence has to be issued. It consists of a lowering of the DATA line

while SCK is high, followed by a low pulse on SCK and raising DATA again while SCK is still high.

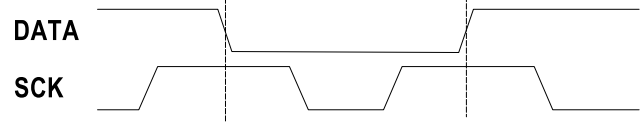


Figure 3 "Transmission Start" sequence

The subsequent command sequence consists of three address bits (only "000" is currently supported) and five command bits. The SHT1x/SHT7x indicates the proper reception of a command by pulling the DATA pin low (ACK bit) after the falling edge of the 8th SCK clock and the DATA line is released (and goes high) after the falling edge of the 9th SCK clock.

See 2.2.5 "Measurement Sequence" for an application of the command sequence

2.2.4 Connection reset sequence

If communication with the device is lost the following signal sequence will reset its serial interface:

While leaving DATA high toggle SCK 9 or more times. This must be followed by a "Transmission Start" sequence preceding the next command. This sequence resets the interface only. The status register preserves its content.

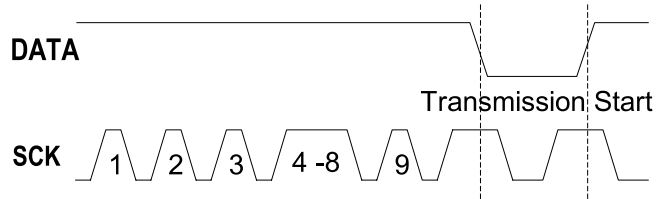
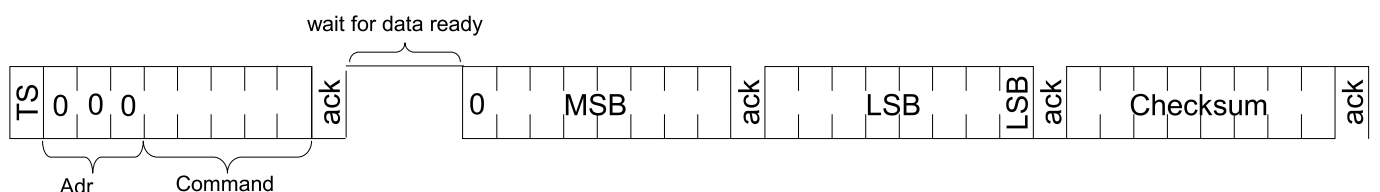


Figure 4 Connection reset sequence

Command	Code	Description
Reserved	0000x	Reserved
Measure Temperature	00011	Temperature measurement
Measure Humidity	00101	Humidity measurement
Status Register Read	00111	Read access to the status register (see application note)
Status Register Write	00110	Write access to the status register (see application note)
Reserved	0101x-1110x	Reserved
Soft reset	11110	resets the chip, clears the status register to default values wait 11ms before next command

Table 2 SHT1x/SHT7x list of commands



2.2.5 Measurement sequence (T and RH)

After issuing a measurement command ('00000101' for RH, '00000011' for Temperature) the controller has to wait for the measurement to complete. This takes approximately 11/55/210ms for a 8/12/14bit measurement. The exact time varies by up to ±15% with the speed of the internal oscillator. To signal the completion of a measurement, the SHT1x pulls down the data line (2) and the controller must restart SCK. Two bytes of measurement data and one byte of CRC checksum will then be transmitted. The uC must acknowledge each byte by pulling the DATA line low. All values are MSB first, right justified. (e.g. the 5th SCK is MSB for a 12bit value, for a 8bit result the first byte is not used). Communication terminates after the acknowledge bit of the

CRC data. If CRC-8 Checksum is not used the controller may terminate the communication after the measurement data LSB by keeping ack high. The device automatically returns to sleep mode after the measurement and communication have finished.

Warning: To keep heat up of the SHT1x/SHT7x below 0.1°C it should not be active for more than 15% of the time (e.g. max. 3 measurements / second for 12bit accuracy).

2.2.6 CRC-8 Checksum Calculation

Please consult application note "CRC-8 Checksum Calculation" for information on how to calculate the CRC.

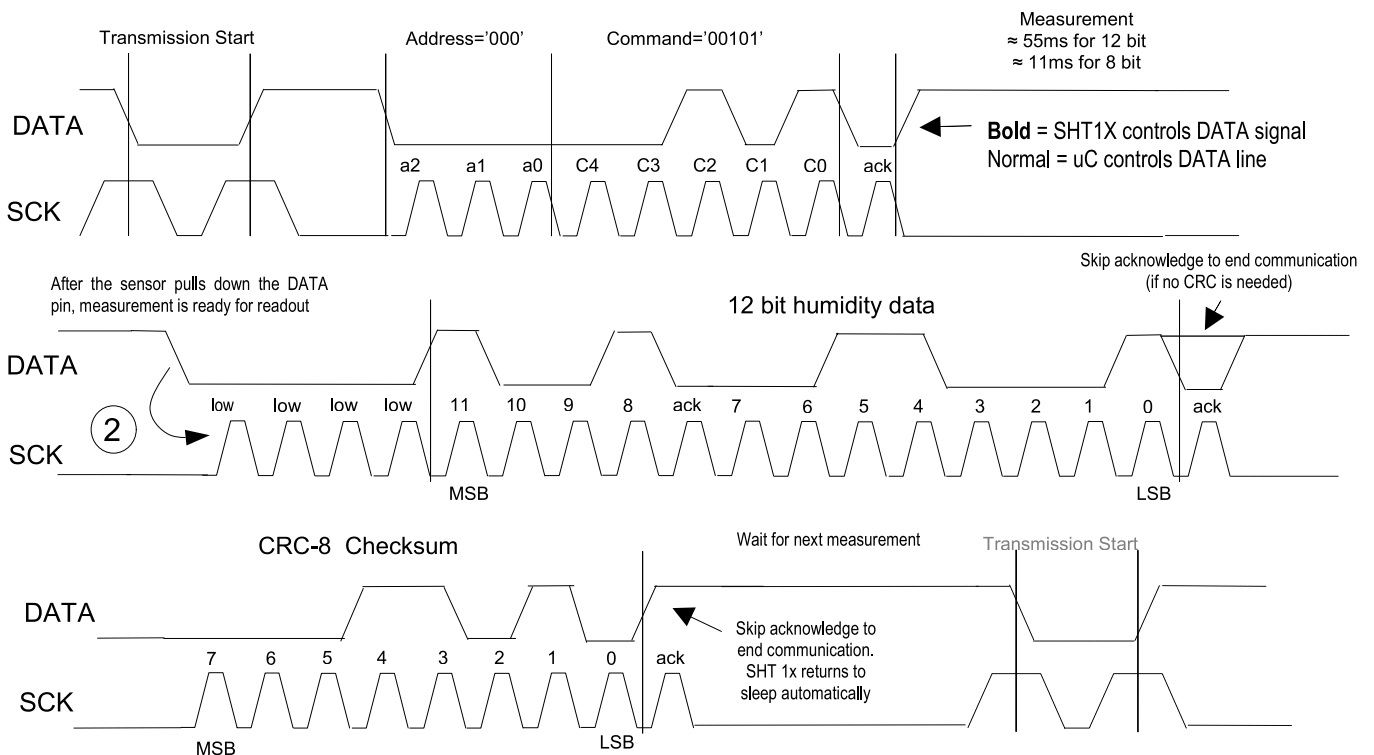


Figure 5 Example RH measurement sequence for value "0000'1001 ' 0011'0001"= 2353 = 75.79%RH

2.3 Status Register

Some of the advanced functions of the SHT1x/SHT7x are available through the status register. The following section gives a brief overview of these features. Please consult application note "Status Register" for more information.

2.3.1 Measurement resolution

The default measurement resolution of 14bit (temperature) and 12bit (humidity) can be reduced to 12 and 8 bit. This is especially useful in high speed or extreme low power applications.

2.3.2 End of Battery

The "End of Battery" function detects VDD voltages below 2.47V. Accuracy is ±0.05V

2.3.3 Heater

An on chip heating element can be switched on. It will increase the temperature of the sensor by approximately 5°C. Power consumption will increase by 8mA @ 5V.

Applications:

- By comparing temperature and humidity values before and after switching on the heater, proper functionality of both sensors can be verified.
- In high RH environments heating the sensor element will avoid condensation.

Warning: The built-in calibration is not correct while the sensmitter is heated!

Please consult application note "Status Register" for more information on how to access and use these features.

4.2 SHT7x (4-pin single-in-line)

SHT7x are available starting Q3 2002

Pin	Name	Comment
1	SCK	Serial clock input
2	VDD	Supply 2.4 – 5.5V
3	GND	Ground
4	DATA	Serial data bidirectional

Table 7 SHT7x Pin Description

4.2.1 Package type

The device is supplied in a single-in-line pin type package. The sensors housing consists of a Liquid Crystal Polymer (LCP) cap with epoxy glob top on a standard 0.6mm FR4 substrate.

The sensor head is connected to the pins by a small bridge to minimize heat conduction and response times.

A 100nF capacitor is mounted on the back side between VDD and GND.

Weight 168mg, Weight of sensor head 73mg

All pins are gold plated to avoid corrosion.

Pins mate with most 1.27mm (0.05") sockets e.g.: Preci-dip / Mill-Max 851-93-004-20-001 or similar

4.2.2 Soldering Information (Preliminary)

The SHT7x may be soldered using standard wave soldering systems at maximum 225°C for 20 seconds.

For manual soldering contact time must be limited to 5 seconds at up to 350°C.

After soldering the devices must be stored at >74%RH for at least 24h to allow the polymer to recover.

Please consult the application note "Soldering procedure" for detailed instructions.

4.2.3 Delivery Conditions

The SHT7x will be delivered in trays by xx pieces per tray. Other delivery options may be available on request.

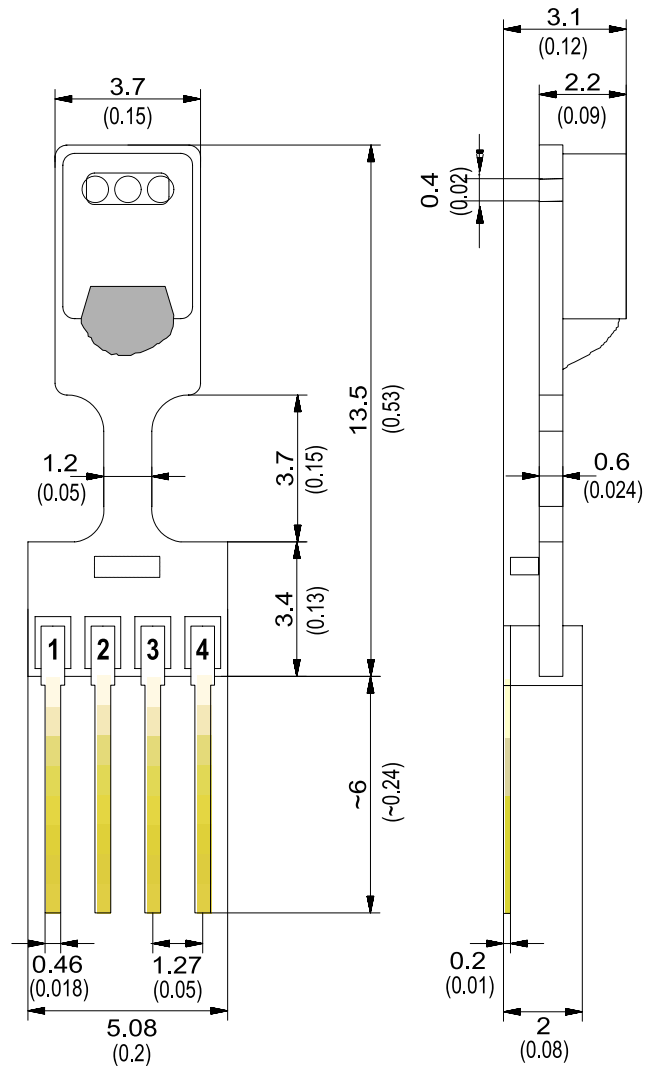


Figure 10 SHT7x dimensions in mm (inch), preliminary

4.3 Other Packages

Other packaging options may be available on request.

4.4 Production date

The production date is printed onto the cap in white numbers in the form wwy. e.g. "351" = week 35, 2001.

4.5 Mounting Recommendations

The relative humidity of a gas strongly depends on its temperature. It is therefore essential to keep the sensor at the same temperature as the air of which the humidity is to be measured.

If the sensmitter shares a PCB with heating electronic components it should be mounted below the heat source and the housing must remain well ventilated. To reduce

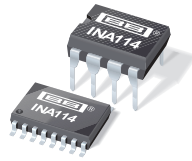
heat conduction copper layers between the SHT1x and the rest of the PCB should be minimized and a slit may be milled in between.

Prolonged direct exposure of the SHT1x/SHT7x to sunshine or UV radiation should be avoided.

4.6 Wiring considerations and signal integrity

Carrying the SCK and DATA signal parallel and in close proximity (e.g. in wires) for more than 10cm may result in cross talk and loss of communication. This may be resolved by routing VDD and/or GND between the two data signals.

Please see the application note "ESD, Latchup and EMC" for more information.



INA114

Precision INSTRUMENTATION AMPLIFIER

FEATURES

- **LOW OFFSET VOLTAGE:** 50 μ V max
- **LOW DRIFT:** 0.25 μ V/ $^{\circ}$ C max
- **LOW INPUT BIAS CURRENT:** 2nA max
- **HIGH COMMON-MODE REJECTION:** 115dB min
- **INPUT OVER-VOLTAGE PROTECTION:** \pm 40V
- **WIDE SUPPLY RANGE:** \pm 2.25 to \pm 18V
- **LOW QUIESCENT CURRENT:** 3mA max
- **8-PIN PLASTIC AND SOL-16**

APPLICATIONS

- BRIDGE AMPLIFIER
- THERMOCOUPLE AMPLIFIER
- RTD SENSOR AMPLIFIER
- MEDICAL INSTRUMENTATION
- DATA ACQUISITION

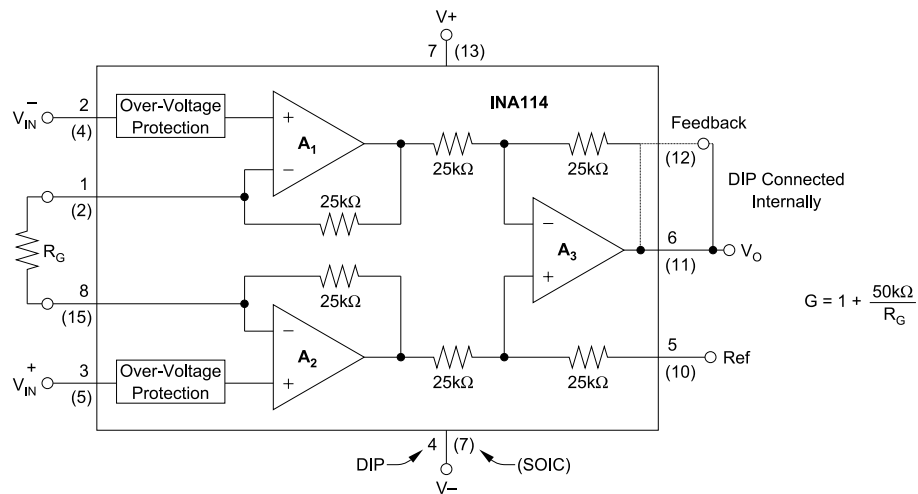
DESCRIPTION

The INA114 is a low cost, general purpose instrumentation amplifier offering excellent accuracy. Its versatile 3-op amp design and small size make it ideal for a wide range of applications.

A single external resistor sets any gain from 1 to 10,000. Internal input protection can withstand up to \pm 40V without damage.

The INA114 is laser trimmed for very low offset voltage (50 μ V), drift (0.25 μ V/ $^{\circ}$ C) and high common-mode rejection (115dB at $G = 1000$). It operates with power supplies as low as \pm 2.25V, allowing use in battery operated and single 5V supply systems. Quiescent current is 3mA maximum.

The INA114 is available in 8-pin plastic and SOL-16 surface-mount packages. Both are specified for the -40° C to $+85^{\circ}$ C temperature range.



International Airport Industrial Park • Mailing Address: PO Box 11400, Tucson, AZ 85734 • Street Address: 6730 S. Tucson Blvd., Tucson, AZ 85706 • Tel: (520) 746-1111 • Twx: 910-952-1111
Internet: <http://www.burr-brown.com/> • FAXLine: (800) 548-6133 (US/Canada Only) • Cable: BBRCORP • Telex: 066-6491 • FAX: (520) 889-1510 • Immediate Product Info: (800) 548-6132

SPECIFICATIONS

ELECTRICAL

At $T_A = +25^\circ\text{C}$, $V_S = \pm 15\text{V}$, $R_L = 2\text{k}\Omega$, unless otherwise noted.

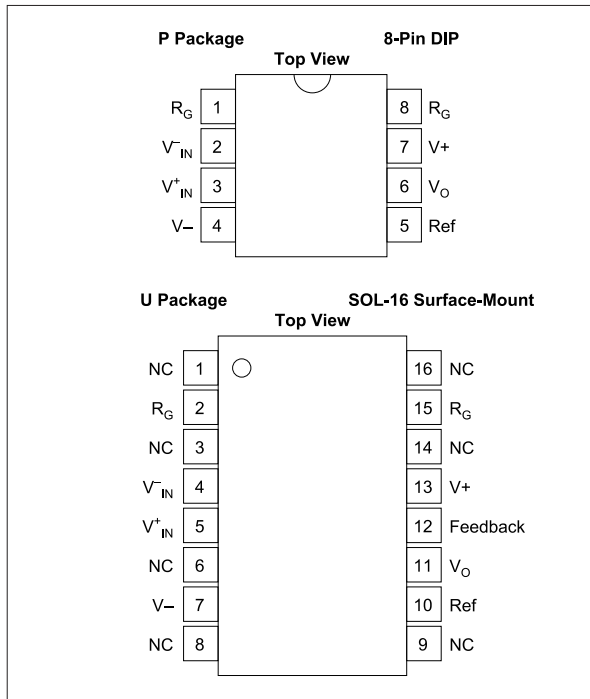
PARAMETER	CONDITIONS	INA114BP, BU			INA114AP, AU			UNITS	
		MIN	TYP	MAX	MIN	TYP	MAX		
INPUT Offset Voltage, RTI Initial vs Temperature vs Power Supply Long-Term Stability Impedance, Differential Common-Mode Input Common-Mode Range Safe Input Voltage Common-Mode Rejection	$T_A = +25^\circ\text{C}$ $T_A = T_{\text{MIN}}$ to T_{MAX} $V_S = \pm 2.25\text{V}$ to $\pm 18\text{V}$ $V_{\text{CM}} = \pm 10\text{V}$, $\Delta R_S = 1\text{k}\Omega$ $G = 1$ $G = 10$ $G = 100$ $G = 1000$		$\pm 10 + 20/G$ $\pm 0.1 + 0.5/G$ $0.5 + 2/G$ $\pm 0.2 + 0.5/G$ $10^{10} \parallel 6$ $10^{10} \parallel 6$	$\pm 50 + 100/G$ $\pm 0.25 + 5/G$ $3 + 10/G$		$\pm 25 + 30/G$ $\pm 0.25 + 5/G$ *	$\pm 125 + 500/G$ $\pm 1 + 10/G$ *	μV $\mu\text{V}/^\circ\text{C}$ $\mu\text{V}/\text{V}$ $\mu\text{V}/\text{mo}$ $\Omega \parallel \text{pF}$ $\Omega \parallel \text{pF}$ V	
			± 11	± 13.5	± 40	*	*	*	
			80	96		75	90		dB
			96	115		90	106		dB
			110	120		106	110		dB
	115	120		106	110		dB		
BIAS CURRENT vs Temperature			± 0.5 ± 8	± 2		*	± 5	nA $\text{pA}/^\circ\text{C}$	
OFFSET CURRENT vs Temperature			± 0.5 ± 8	± 2		*	± 5	nA $\text{pA}/^\circ\text{C}$	
NOISE VOLTAGE, RTI $f = 10\text{Hz}$ $f = 100\text{Hz}$ $f = 1\text{kHz}$ $f_B = 0.1\text{Hz}$ to 10Hz Noise Current $f = 10\text{Hz}$ $f = 1\text{kHz}$ $f_B = 0.1\text{Hz}$ to 10Hz	$G = 1000$, $R_S = 0\Omega$		15 11 11 0.4			*	*	$\text{nV}/\sqrt{\text{Hz}}$ $\text{nV}/\sqrt{\text{Hz}}$ $\text{nV}/\sqrt{\text{Hz}}$ $\mu\text{Vp-p}$	
				0.4 0.2 18			*	*	$\text{pA}/\sqrt{\text{Hz}}$ $\text{pA}/\sqrt{\text{Hz}}$ pAp-p
			1	$1 + (50\text{k}\Omega/R_G)$	10000	*	*	*	V/V V/V
			$G = 1$ $G = 10$ $G = 100$ $G = 1000$	± 0.01 ± 0.02 ± 0.05 ± 0.5	± 0.05 ± 0.4 ± 0.5 ± 1	*	*	*	% % % %
			$G = 1$ $G = 10$ $G = 100$ $G = 1000$	± 2 ± 25 ± 0.0001 ± 0.0005 ± 0.0005 ± 0.002	± 10 ± 100 ± 0.001 ± 0.002 ± 0.002 ± 0.01	*	*	*	$\text{ppm}/^\circ\text{C}$ $\text{ppm}/^\circ\text{C}$ % of FSR % of FSR % of FSR % of FSR
OUTPUT Voltage Load Capacitance Stability Short Circuit Current	$I_O = 5\text{mA}$, T_{MIN} to T_{MAX} $V_S = \pm 11.4\text{V}$, $R_L = 2\text{k}\Omega$ $V_S = \pm 2.25\text{V}$, $R_L = 2\text{k}\Omega$	± 13.5 ± 10 ± 1	± 13.7 ± 10.5 ± 1.5 1000 $+20/-15$		*	*	*	V V V pF mA	
						*	*	*	
						*	*	*	
FREQUENCY RESPONSE Bandwidth, -3dB Slew Rate Settling Time, 0.01% Overload Recovery	$G = 1$ $G = 10$ $G = 100$ $G = 1000$ $V_O = \pm 10\text{V}$, $G = 10$ $G = 1$ $G = 10$ $G = 100$ $G = 1000$ 50% Overdrive		1 100 10 1 0.6 18 20 120 1100 20			*	*	MHz kHz kHz kHz V/ μs μs μs μs μs μs	
			0.3			*	*	*	
						*	*	*	
						*	*	*	
						*	*	*	
POWER SUPPLY Voltage Range Current	$V_{\text{IN}} = 0\text{V}$	± 2.25	± 15 ± 2.2	± 18 ± 3	*	*	*	V mA	
					*	*	*		
TEMPERATURE RANGE Specification Operating θ_{JA}		-40 -40		85 125	*	*	*	$^\circ\text{C}$ $^\circ\text{C}$ $^\circ\text{C}/\text{W}$	
			80		*	*	*		
						*	*		

* Specification same as INA114BP/BU.

NOTE: (1) Temperature coefficient of the "50k Ω " term in the gain equation.

The information provided herein is believed to be reliable; however, BURR-BROWN assumes no responsibility for inaccuracies or omissions. BURR-BROWN assumes no responsibility for the use of this information, and all use of such information shall be entirely at the user's own risk. Prices and specifications are subject to change without notice. No patent rights or licenses to any of the circuits described herein are implied or granted to any third party. BURR-BROWN does not authorize or warrant any BURR-BROWN product for use in life support devices and/or systems.

PIN CONFIGURATIONS



ELECTROSTATIC DISCHARGE SENSITIVITY

This integrated circuit can be damaged by ESD. Burr-Brown recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

PACKAGE/ORDERING INFORMATION

PRODUCT	PACKAGE	PACKAGE DRAWING NUMBER ⁽¹⁾	TEMPERATURE RANGE
INA114AP	8-Pin Plastic DIP	006	-40°C to +85°C
INA114BP	8-Pin Plastic DIP	006	-40°C to +85°C
INA114AU	SOL-16 Surface-Mount	211	-40°C to +85°C
INA114BU	SOL-16 Surface-Mount	211	-40°C to +85°C

NOTE: (1) For detailed drawing and dimension table, please see end of data sheet, or Appendix C of Burr-Brown IC Data Book.

ABSOLUTE MAXIMUM RATINGS⁽¹⁾

Supply Voltage	±18V
Input Voltage Range	±40V
Output Short-Circuit (to ground)	Continuous
Operating Temperature	-40°C to +125°C
Storage Temperature	-40°C to +125°C
Junction Temperature	+150°C
Lead Temperature (soldering, 10s)	+300°C

NOTE: (1) Stresses above these ratings may cause permanent damage.

APPLICATION INFORMATION

Figure 1 shows the basic connections required for operation of the INA114. Applications with noisy or high impedance power supplies may require decoupling capacitors close to the device pins as shown.

The output is referred to the output reference (Ref) terminal which is normally grounded. This must be a low-impedance connection to assure good common-mode rejection. A resistance of 5Ω in series with the Ref pin will cause a typical device to degrade to approximately 80dB CMR ($G = 1$).

SETTING THE GAIN

Gain of the INA114 is set by connecting a single external resistor, R_G :

$$G = 1 + \frac{50 \text{ k}\Omega}{R_G} \quad (1)$$

Commonly used gains and resistor values are shown in Figure 1.

The 50kΩ term in equation (1) comes from the sum of the two internal feedback resistors. These are on-chip metal film resistors which are laser trimmed to accurate absolute val-

ues. The accuracy and temperature coefficient of these resistors are included in the gain accuracy and drift specifications of the INA114.

The stability and temperature drift of the external gain setting resistor, R_G , also affects gain. R_G 's contribution to gain accuracy and drift can be directly inferred from the gain equation (1). Low resistor values required for high gain can make wiring resistance important. Sockets add to the wiring resistance which will contribute additional gain error (possibly an unstable gain error) in gains of approximately 100 or greater.

NOISE PERFORMANCE

The INA114 provides very low noise in most applications. For differential source impedances less than 1kΩ, the INA103 may provide lower noise. For source impedances greater than 50kΩ, the INA111 FET-input instrumentation amplifier may provide lower noise.

Low frequency noise of the INA114 is approximately 0.4μVp-p measured from 0.1 to 10Hz. This is approximately one-tenth the noise of "low noise" chopper-stabilized amplifiers.

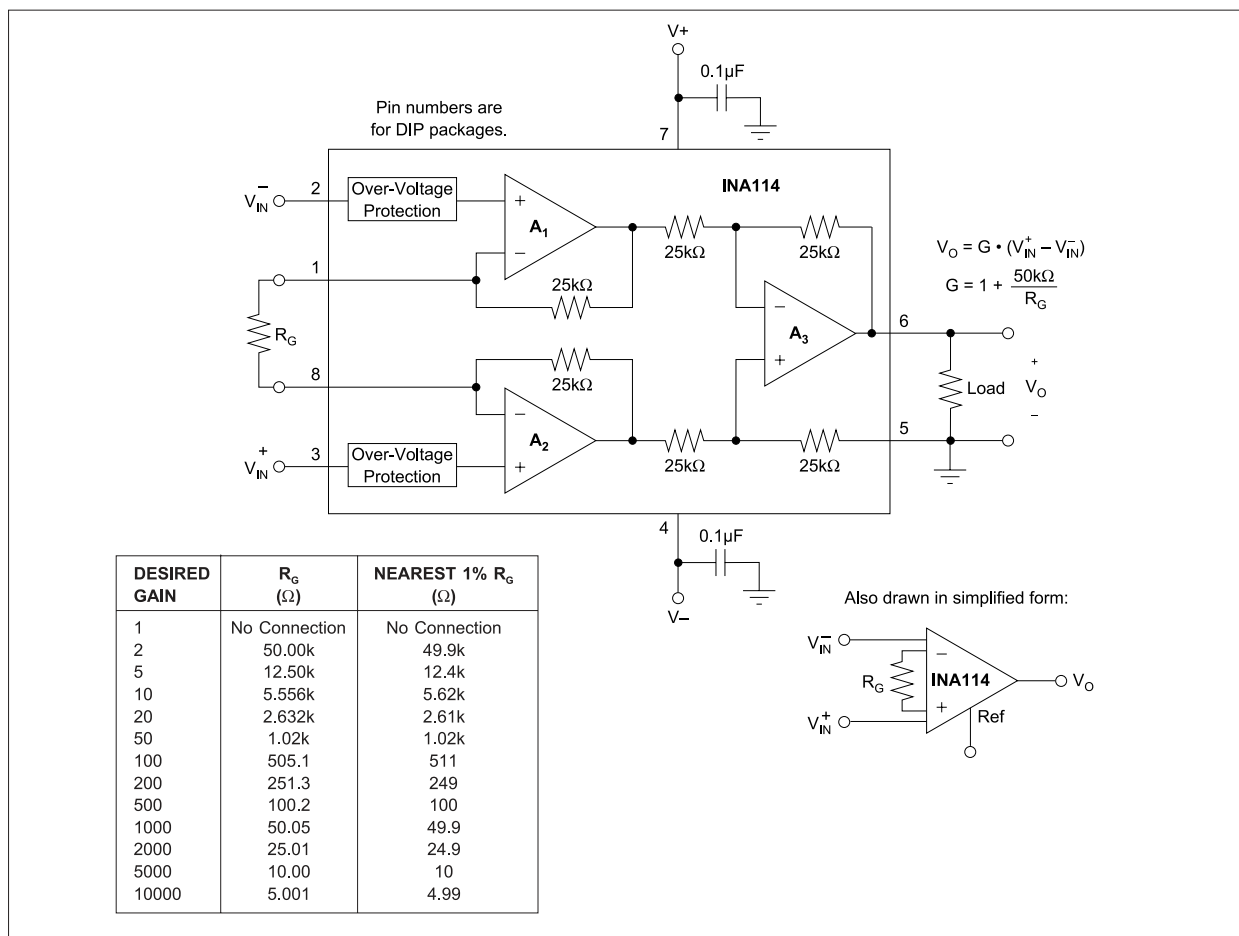


FIGURE 1. Basic Connections.

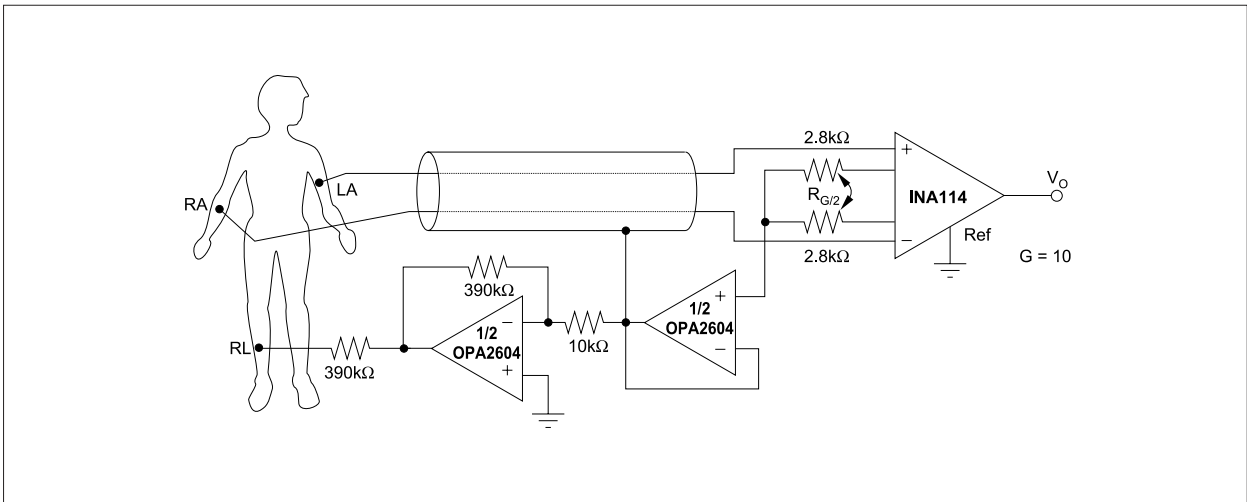


FIGURE 10. ECG Amplifier With Right-Leg Drive.

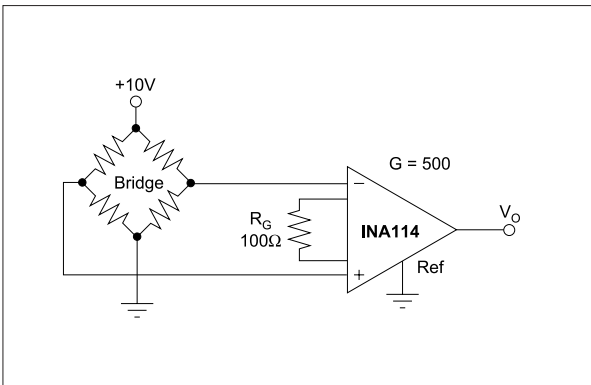


FIGURE 11. Bridge Transducer Amplifier.

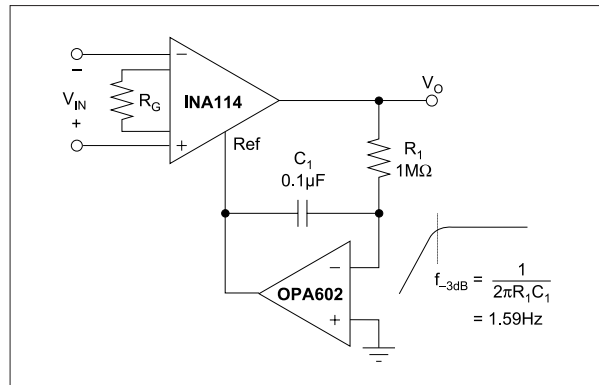


FIGURE 12. AC-Coupled Instrumentation Amplifier.

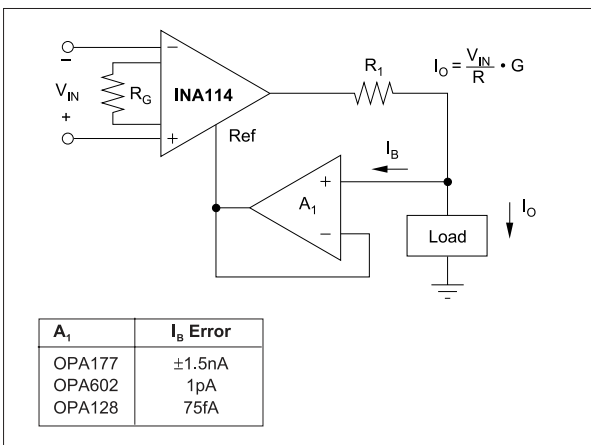


FIGURE 13. Differential Voltage-to-Current Converter.