**Note :** Ceci est simplement « un modèle » à compléter selon vos soins. Des adaptations sont autorisées à condition de les justifier. **Ecoutez les conseils de votre parrain.**

**Première partie : AVOIR UNE VUE D’ENSEMBLE DU PROJET**

1. ***Dessinez l’architecture du projet – comment avez-vous compris le projet ?***

1. Premier montage Arduino

a) Récupérer le matériel

b) Refaire le montage

2. Premier programme Arduino

-> Détecte, mesure, calcule le pouls

a) main.c

-> Loop et setup

b) cardio.c/.h

-> Fonctions récolte informations/calcul pouls

3. Deuxième montage Arduino

a) Récupérer le matériel

b) Faire le montage

4. Deuxième programme Arduino

a) coeur.c/coeur.h

-> LEDs allumées en même temps au rythme des

battements de coeur détectés par le module Cardio.

b) param.h (Après programme C)

-> configuration allumage des LEDs (2e montage)

5. programme C

-> création du param.h

a) main.c

-> Lance le menu en console

b) menu.c/.h

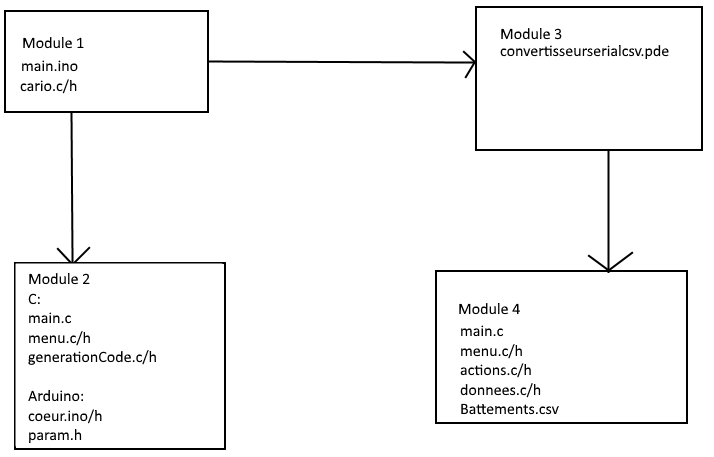
-> fonctions/création menus de paramétrage du coeur de LEDs

c) generationCode.c/.h

-> fonctions pour créer le fichier param.h

**Deuxième partie : ANALYSEZ LES STRUCTURES DE DONNEES DU PROJET**

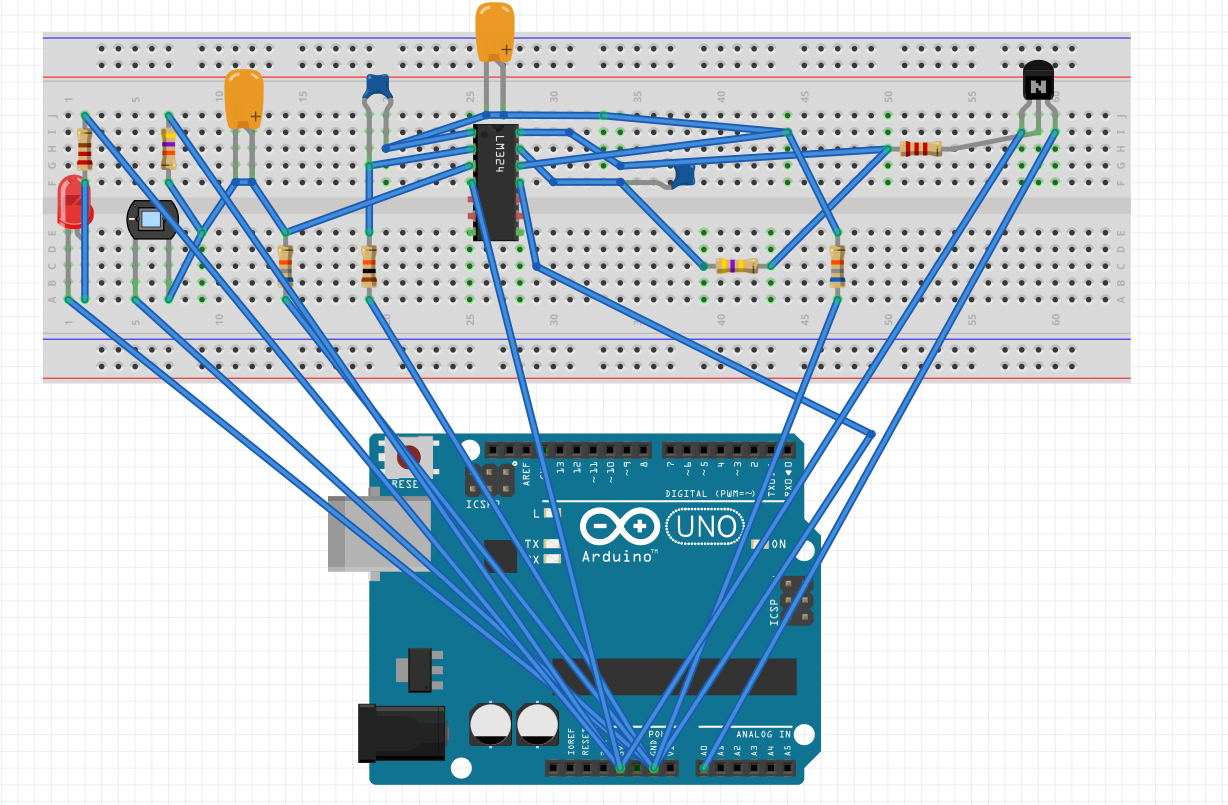
***1. Représentation graphique de toutes les structures nécessaires, organisation des fichiers de code Arduino et du code C et dépendances entre les fichiers.***



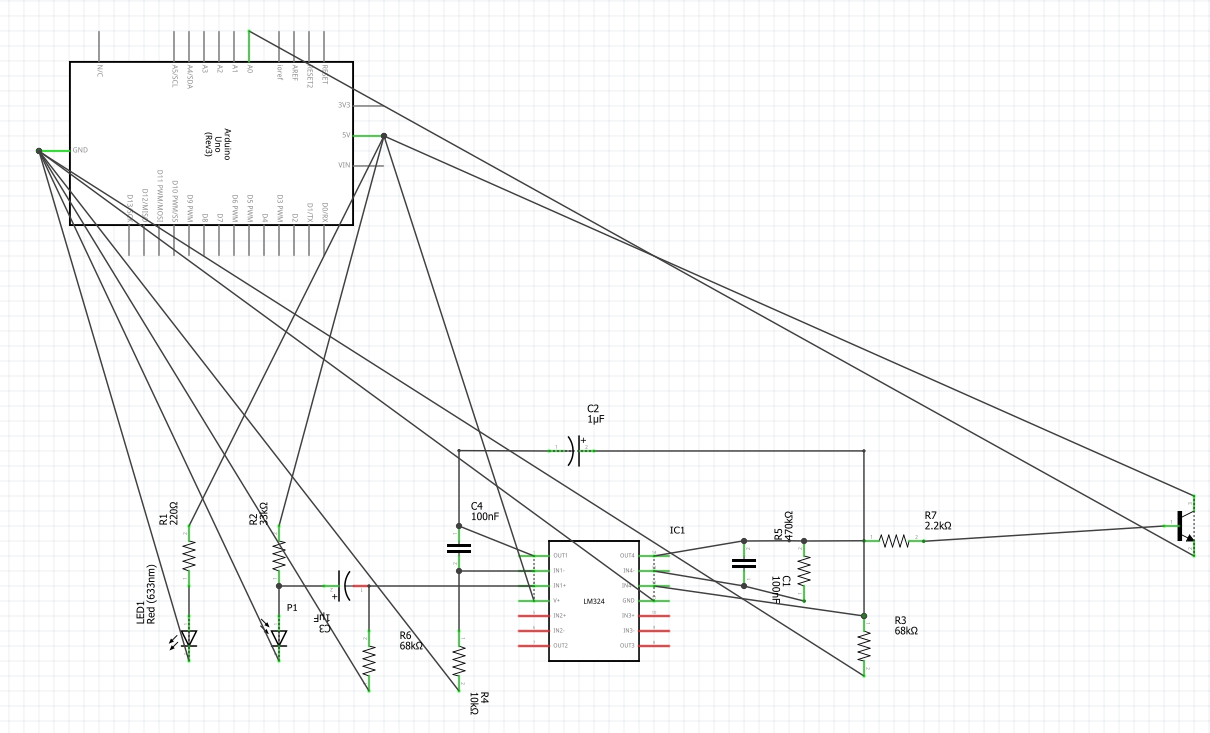
**Troisième partie : MODULARISATION & WORKFLOW DE FONCTIONS & SCHEMAS ELECTONIQUES**

1. ***Schéma électroniques avec les composants sur Fritzing (vue platine et vue schématique des module cardio et cœur de LEDs ). Comme cette partie comporte une évaluation séparée du projet, vous pouvez faire un document à part entière.***

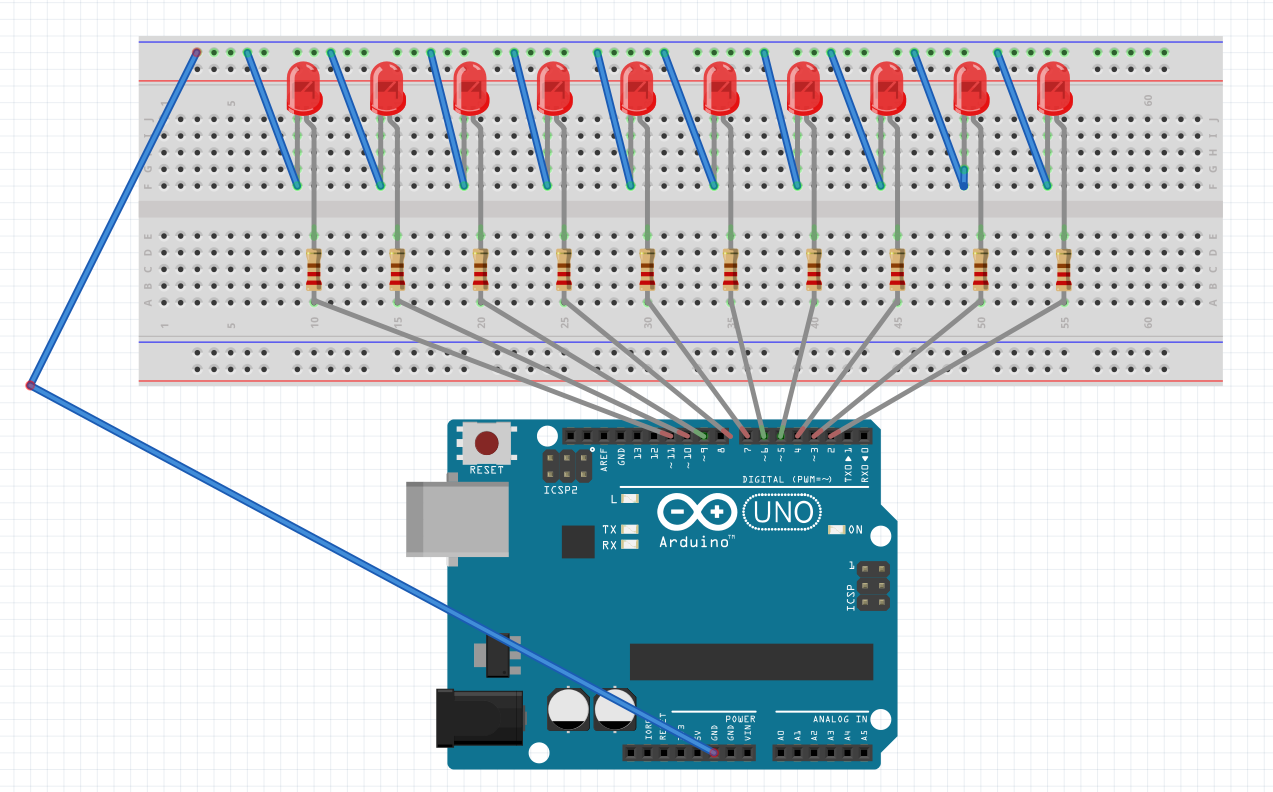
**Pouls vue platine :**



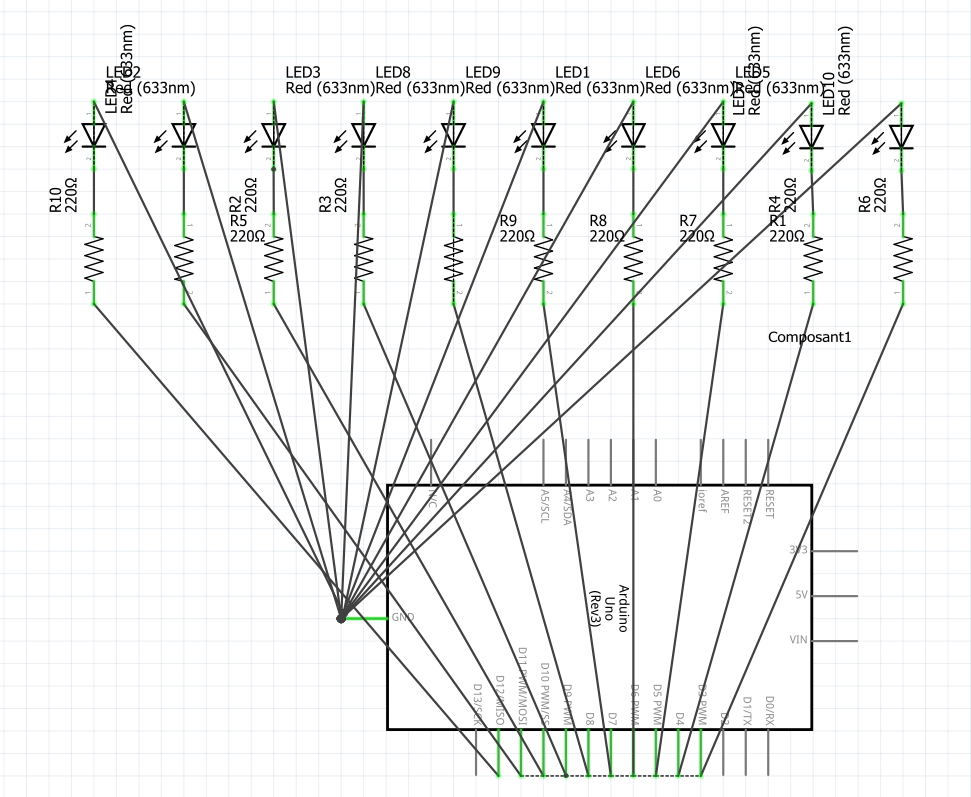
**Pouls vue schématique :**



**Cœur vue platine :**



**Cœur vue schématique :**



1. ***Description algorithmique chaque partie du projet (module cardio, module cœur de LEDs (inclus la génération automatique du paramétrage à partir d’un programme en C), module Preprocessing/acquisition des données, module lecture et traitement de données en C)***

Le module Cardio récupère le pouls en analysant la valeur renvoyée par la LED IR lors d’une pulsation, en incrémentant la variable du pouls.

Le module Cœur analyse un tableau dans un tableau pour avoir une suite de données à afficher grâce aux LEDs. Le param.h délimite la suite à afficher. Ce header est généré par un programme en C qui demande les valeurs de début et de fin (qui délimitent la suite à afficher), puis écrit tout le code situé dans le param.h.

Si le param.h existe déjà, il est remplacé.

Le module Preprocessing récupère les valeurs du module Cardio et les écrit dans un fichier csv.

Le module Lecture et traitement des données lit les données se trouvant dans le fichier csv, puis les traite de différentes manières. Il peut les trier avec une fonction de tri par insertion. Il peut chercher des valeurs spécifiques grâce à une fonction de recherche par dichotomie. Il peut calculer la moyenne des valeurs. Il peut également les afficher sans aucun traitement.

1. ***Prototypes de l'ensemble des fonctions du projet (faites-le le plus « graphiquement » possible)***

**Module 1 :**

Cardio

void calculPulsation(int valeurActuelle, int valeurSeuil, long tempsDetection);

**Module 2 :**

Génération code

int ouvrirFichier(int f);

void ecrireFichier(f,DEBUT,FIN) ;

int ouvrirFichier(int f) ;

Menu

int menu(int f);

**Module 4 :**

Action

void average(Value \*tab, int numberLine, long val1, long val2);

int dichoSearch(Value \*tab, int numberLine, long val);

void sortInsertion(Value \*tab, int numberLine, int valueSort, int typeSort);

void display(Value \*tab, int numberLine);

void treatment(Value \*tab, int \*menuChoice, int \*numberLine);

Menu

void menuSort(int menuSortChoice);

Données

void getValue(Value \*tab, FILE\* content);

void getSize(FILE\* content, int\* numberLine);

**Quatrième partie : REPARTISSEZ-VOUS LES TACHES**

|  |  |
| --- | --- |
| **Nom : Pierre-Alain** | **Rôle principal : Programmation Arduino** |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tâches** | **10/11** | **11/11** | **12/11** | **13/11** | **14/11** | **15/11** | **16/11** | **17/11** | **18/11** | **19/11** | **20/11** |
| Module 1 |  |  |  | Ok | Ok | Ok | Fini | Fini |  |  |  |
| Module 2 |  |  |  | Non | Ok | Ok | Fini | Fini |  |  |  |
| Module 4 |  |  |  | Non | Non | Non | Ok | Ok |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Nom : Hugo** | **Rôle principal : Programmation Processing et montages Arduino** |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tâches** | **10/11** | **11/11** | **12/11** | **13/11** | **14/11** | **15/11** | **16/11** | **17/11** | **18/11** | **19/11** | **20/11** |
| Module 1 |  |  |  | Ok | Ok | Fini | Fini | Fini |  |  |  |
| Module 2 |  |  |  | Non | Ok | Ok | Fini | Fini |  |  |  |
| Module 4 |  |  |  | Non | Non | Non | Ok | Ok |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **Nom : François** | **Rôle principal : Programmation création param.h** |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tâches** | **10/11** | **11/11** | **12/11** | **13/11** | **14/11** | **15/11** | **16/11** | **17/11** | **18/11** | **19/11** | **20/11** |
| Module 2 |  |  |  | Abs | Abs | Abs | Ok | Fini |  |  |  |
| Module 4 |  |  |  | Abs | Abs | Abs | Ok | Ok |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |