

# **Projet final TSO**

**Contrôleur d'écran**

**Programme de TSO**

**Cours :**

**247-627-LI :**

**Analyser et caractériser un système ordonné**

**Présenté par :**

**Charles Lévesque-Matte  
Pascal-Emmanuel Lachance**

**Présenté à :**

**Simon Ayoub**

**Date : 5 Novembre 2018**

**Cégep Limoilou**

## Table des matières

<b>1. DÉFINITION DU PROJET .....</b>	<b>3</b>
1.1 CONTEXTE: .....	3
1.2 OBJECTIF DU PROJET: .....	4
1.3 DESCRIPTION DU PRODUIT INCLUANT LA LISTE DES LIVRABLES DU PROJET : .....	4
1.4 CONTRAINTES : .....	4
1.5 GESTION DES RISQUES : .....	5
1.5.1 Alimentation : .....	5
1.5.2 Conception du PCB : .....	7
1.5.3 Conception du boîtier : .....	9
1.5.4 Erreur de programmation : .....	10
1.5.5 Manque de documentation : .....	10
1.5.6 Communication trop lente avec l'écran : .....	11
1.5.7 Manque de temps : .....	11
1.5.8 Quantité de portes logiques : .....	12
<b>2 LISTE DES REQUIS .....</b>	<b>13</b>
2.1 LISTE DES REQUIS MATÉRIELS .....	14
2.2 LISTE DES REQUIS ÉLECTRIQUES .....	14
2.3 LISTE DES REQUIS LOGICIELS .....	15
<b>3 SCHÉMAS BLOC PHYSIQUES .....</b>	<b>16</b>
3.1 SCHÉMA BLOC GLOBAL – NIVEAU 0 : .....	16
3.1.1 SCHÉMA BLOC GLOBAL – NIVEAU 1 : .....	17
3.1.1.1 SCHÉMA BLOC ALIMENTATION – NIVEAU 2 : .....	18
3.1.1.2 SCHÉMA BLOC CONNECTEUR 40-BROCHES – NIVEAU 2 : .....	19
3.1.1.3 SCHÉMA BLOC CONNECTEUR USB – NIVEAU 2 : .....	20
3.1.1.4 SCHÉMA BLOC CONNECTEUR PC – NIVEAU 2 .....	21
3.1.1.5 SCHÉMA BLOC CONNECTEUR ÉCRAN – NIVEAU 2 .....	22
3.1.1.6 SCHÉMA BLOC DALLE TACTILE – NIVEAU 2 .....	23
3.1.1.7 SCHÉMA BLOC FPGA – NIVEAU 2 .....	24
3.1.1.1.1 SCHÉMA BLOC ALIMENTATION À DÉCOUPAGE – NIVEAU 3 .....	25
<b>4 PLANIFICATION DU PROJET .....</b>	<b>26</b>
4.1 DIAGRAMMES WBS.....	26
4.1.1 Documentation & Gestion .....	26
4.1.2 Alimentation.....	27
4.1.3 Communication.....	27
4.1.4 Traitement externe .....	28
4.1.5 Traitement interne.....	29
4.1.6 Pavé tactile.....	29
4.1.7 Affichage .....	30
4.1.8 PCB & Boîtier.....	30
4.2 DIAGRAMME DE GANTT.....	31

## I. Définition du projet

### I.1 Contexte:

Il y a quelques années déjà, le collège a reçu des écrans monochromes de la part d'une compagnie les ayant achetés auparavant. Cependant, ces écrans n'ayant aucun contrôleur, ils n'en trouvèrent d'utilité. Ils en firent alors le don au département de génie électrique du Cégep Limoilou, sachant que ces passionnés de l'électronique y trouveraient leur compte. L'enseignant Daniel Côté prit en main le projet et le proposa à des étudiants finissants en 2011. Le projet n'aboutit pas, ni en 2011, ni dans les années qui suivirent, c'est donc sur nos frêles épaules que retombe cette tâche colossale et extrêmement gratifiante, puisqu'elle laissera une marque auprès des prochaines générations d'étudiants en T.S.O. En effet, ce projet, une fois mené à terme, pourra être utilisé à des fins de laboratoire. Ce projet nous intéresse puisqu'il requiert des connaissances que nous n'avons guère développées durant notre cursus jusqu'à maintenant. Le contrôleur d'écran nous motive puisqu'il nous permettra de participer au programme de T.S.O. en laissant un outil d'apprentissage, et ce même après notre passage. La conception de circuit à l'aide d'un FPGA nous intéresse, car il permet de faire un pont entre l'électronique et la programmation. En effet, ce lien est une des raisons pour lesquelles nous avons rejoint le programme en premier lieu et la technologie FPGA permet de pousser cette connexion à son paroxysme. Nous sommes motivés par un but commun, l'apprentissage de nouvelles connaissances : la conception FPGA, le développement de projet dans son intégralité, support logiciel avec des bibliothèques, création d'un protocole de communication efficace et compatible avec plusieurs appareils et la gestion de projet (temps, ressources, risques).

## 1.2 Objectif du projet:

La finalité du projet se verra être un contrôleur d'écran fonctionnant sur les cartes DALLAS, STM et possiblement d'autres cartes de développement.

## 1.3 Description du produit incluant la liste des livrables du projet :

Le produit se veut être un contrôleur vidéo incluant un écran, un FPGA, une alimentation avec plusieurs tensions de sorties, un pavé tactile et plusieurs entrées. L'ensemble sera interfacé aux cartes de développement DALLAS et STM32 par communication parallèle, I2C et RS-232. L'écran sera collé sur le circuit imprimé de façon à être le plus près possible de ce qu'un manufacturier pourrait produire. Un pavé tactile capacitif sera monté sur l'écran, et aura un contrôleur fourni, ce qui le rendra par conséquent plus facile à interfacé avec le reste du système. Notre surface d'affichage supportera non seulement l'écriture de caractères matriciels, mais permettra également de visualiser des images à une fréquence de rafraîchissement suffisante, qui donnera alors l'impression d'un flux vidéo continu.

## 1.4 Contraintes :

Nous avons plusieurs contraintes, incluant certaines de types mécaniques, électriques et logiciels. En ce qui a trait aux contraintes mécaniques, il faut notamment que l'unique PCB du projet puisse être contenu dans une boîte, et qu'il permette de nombreuses connexions grâce à son connecteur 40-pins servant à une communication parallèle bidirectionnelle, un port USB pour du RS-232 et un connecteur « Molex » utilisé pour la communication PC. Il faudra que le PCB et l'écran qui y sera collé puissent être placés sur le dessus de la boîte, possiblement dans une encoche dans le boîtier, placée stratégiquement pour qu'on puisse y placer l'écran, accompagné de sa dalle tactile. Du côté des contraintes électriques, de nombreux niveaux d'alimentation doivent être disponibles pour le FPGA, l'écran lui-même et les diverses composantes du système. Les protocoles de communications mentionnés plus haut devront également se faire avec des paramètres précis, mentionnés dans la section 2.2 Liste des requis électriques. Logiciellement, des bibliothèques pour les kits Dallas et STM32 devront être fournies pour écrire sur l'écran, autant en mode texte que graphique, via les différents protocoles, et lire les informations de la dalle tactile.

## **I.5 Gestion des risques :**

### **I.5.1 Alimentation :**

Les risques inhérents à l'alimentation sont représentés par l'impact, la criticité ainsi que le niveau de probabilité qu'ils surviennent durant la réalisation de notre projet. Si l'alimentation ne fournit pas suffisamment de courant, il pourrait en résulter une malfonction de l'écran et une surchauffe de certaines composantes. C'est pourquoi nous allons nous assurer que tous nos calculs tiennent compte des valeurs maximales des composantes et le choix de ces dernières sera fait en conséquence. Il se peut également que nous ayons une mauvaise séquence d'alimentation, par exemple : les entrées-sorties doivent être alimentées avant le cœur ou vice versa, auquel cas la lecture des diagrammes de temps de chaque composantes et la compréhension de notre alimentation sont des éléments primordiaux. Comme dans tout circuit électronique, le bruit et les interférences sont des ennemis sournois pour lesquels nous devons nous préparer convenablement, par exemple en éloignant le circuit d'alimentation des lignes de données sensibles au bruit, et peut-être même en recouvrant les composants sensibles d'une cage de Faraday. Une alimentation instable n'est pas souhaitable, c'est pourquoi nous calculerons le courant demandé et planifierons des composantes plus robustes.

N°	Description	Catégorie	Probabilité	Impact	Criticité	Mesure d'atténuation	Plan de réponse
1	L'alimentation ne fournit pas assez de courant	Main d'œuvre	Faible	Mineur	Négligeable	Calculer avec précision le courant demandé	Source d'alimentation externe
2	Mauvaise séquence d'alimentation	Méthode	Moyenne	Majeur	Significative	Adapter la séquence en fonction de la documentation	Ajout de composante discrète pour temporiser son action
3	Bruit électromagnétique	Milieu	Forte	Modéré	Élevée	Conception sécuritaire (cage de Faraday, éloigner les signaux de l'alimentation)	Circuit de filtration
4	Alimentation instable	Matière	Moyenne	Mineur	Modérée	Mettre des condensateurs de découplage	Augmenter la capacitance de ces derniers

Figure 1 - Alimentation : Table des risques

### **I.5.2 Conception du PCB :**

Lors de la conception de notre PCB, plusieurs facteurs seront critiques et nous devons faire attention de placer nos composantes afin de respecter certaines contraintes. Par exemple, dans la conception d'un pilote d'écran, il faut que les délais de propagation d'ondes soient calculés et pris en compte. Nous devons également prendre conscience des interférences électromagnétiques possibles, principalement causées par la proximité des pièces sur la carte et à l'utilisation d'une bobine pour faire augmenter la tension. Nous souhaitons d'avoir un PCB à peine plus grand que l'écran, ce qui donnerait une impression plus professionnelle, et permettrait à la carte d'entrer dans le boîtier. Cette contrainte entraîne le risque de manquer d'espace sur le PCB. Il ne faut guère oublier que nous ne sommes pas encore des technologues chevronnés, c'est-à-dire que nous sommes propices à faire des erreurs de conception. Une bonne lecture de la documentation et une double vérification devraient nous permettre d'éviter ce genre d'erreur coûteuse et frustrante. Dans les requis nous avons déterminé que le PCB devrait s'interfacer avec une carte de développement, il faudra donc un connecteur robuste qui pourra survivre aux tortures que lui feront subir les étudiants en laboratoire. L'emplacement de ce connecteur doit être choisi méticuleusement pour ne pas ajouter à sa fragilité mécanique.

N°	Description	Catégorie	Probabilité	Impact	Criticité	Mesure d'atténuation	Plan de réponse
1	Temps de propagation	Milieu	Faible	Majeur	Significative	Router la carte à la main	
2	Interférences	Milieu	Faible	Modéré	Modérée	L'emploi d'une cage de Faraday.	Filtre
3	Manque de place	Milieu	Forte	Mineur	Significative	Prendre les plus petites pièces possible	Grossir la carte
4	Erreurs de conception	Méthode	Moyenne	Majeur	Élevée	Révision par un pair, bonne lecture des fiches techniques	Faire un nouveau prototype
5	Fragilité mécanique	Matière	Faible	Mineur	Négligeable	Sélection de pièces peu fragile, disposition adéquate	Ajout de support mécanique

Figure 2 - Conception du PCB : Table des risques



### 1.5.3 Conception du boîtier :

Notre boîtier devra tenir compte de la fragilité mécanique du matériau avec laquelle la boîte sera fabriquée. Le boîtier devra avoir suffisamment de place pour contenir le projet, sans toutefois être trop gros. Nous devons inévitablement penser au fait que notre carte risque de chauffer à cause des pièces qui sont montées dessus, ce qui veut dire que notre boîtier doit permettre la circulation d'air. Comme dans bien des boîtiers adaptés à une carte, il y a des risques d'erreurs de conception, par exemple : les supports pour tenir la carte qui seraient de trop petites dimensions.

N°	Description	Catégorie	Probabilité	Impact	Criticité	Mesure d'atténuation	Plan de réponse
1	Fragilité mécanique	Matière	Faible	Modéré	Négligeable	Épaissir les côtés de la boîte	Ajout de support
2	Manque de place	Milieu	Forte	Mineur	Significative	Faire le PCB le plus petit possible	Agrandir la boîte
3	Chauffage	Milieu	Forte	Majeur	Élevée	Trou d'aération	Dissipateur de chaleur
4	Erreurs de conception	Méthode	Moyenne	Modéré	Significative	Révision de conception	

Figure 3 - Conception du boîtier : Table des risques

#### 1.5.4 Erreur de programmation :

Notre projet comporte aussi des risques au niveau de la programmation de la Dallas et de la STM32 lors de l'interaction entre les cartes de développement et l'écran. Il peut y avoir des erreurs de collisions lors des communications, ou des erreurs dans les matrices que nous envoyons à l'écran. Il se pourrait même que nous ayons des erreurs au niveau du FPGA; dans sa conception, il pourrait ne pas tenir compte de certains paramètres et ne plus fonctionner de façon normale lorsque certaines exceptions surviennent.

N°	Description	Catégorie	Probabilité	Impact	Criticité	Mesure d'atténuation	Plan de réponse
1	Collisions	Main d'œuvre	Faible	Majeur	Modérée	Gestion des collisions	Bloquer accès matériel lors de la communication
2	Erreur de matrice	Main d'œuvre	Moyenne	Mineur	Modérée	Reprise du code	---
3	Erreur niveau FPGA	Main d'œuvre	Faible	Modéré	Modérée	Reprise du code	---

Figure 4 - Erreurs de programmation : Table des risques

#### 1.5.5 Manque de documentation :

Étant donné que l'écran avec lequel nous travaillons n'est plus en production, il se pourrait qu'il manque de documentation accessible, auquel cas nous nous trouverons dans une impasse. C'est pourquoi nous avons récupéré les données des équipes précédentes. Dans un cas de conception d'une composante électronique comme celle-ci, la documentation est clé de voûte du projet. Si jamais nous nous trouvons devant un problème majeur auquel les fiches techniques ne donnent pas de solution, nous pourrions toujours faire de la rétro-ingénierie.

N°	Description	Catégorie	Probabilité	Impact	Criticité	Mesure d'atténuation	Plan de réponse
1	Manque de documentation	Matière	Moyenne	Majeur	Élevée	S'assurer avec notre coach que nous avons tout	Faire de l'ingénierie inverse

Figure 5 - Manque de documentation : Table des risques

### 1.5.6 Communication trop lente avec l'écran :

Étant donné que nous voulons supporter plusieurs protocoles de communication avec notre écran, il se peut que nous manquions de vitesse lors de l'affichage. Il faudra bien implémenter notre solution afin que rien n'y paraisse à l'œil humain. Le RS-232 a ses limites, il faudra alors choisir des taux de transfert élevés (par exemple le standard de 115200 bauds) pour que l'œil humain n'y voie aucune différence avec la communication en parallèle. Le protocole SPI ne pose pas de problème à ce niveau-là puisqu'il peut facilement être employé à des vitesses dans les mégahertz.

N°	Description	Catégorie	Probabilité	Impact	Criticité	Mesure d'atténuation	Plan de réponse
1	Communication trop lente	Milieu	Moyenne	Majeur	Significative	Avoir une vitesse suffisante	Mettre un tampon

Figure 6 - Communication trop lente avec l'écran : Table des risques

### 1.5.7 Manque de temps :

Comme dans tout projet, le temps est critique et nous nous devons de respecter les délais impartis au projet. Il se pourrait, par conséquent, que nous manquions de temps. Ce serait une très mauvaise nouvelle pour la réalisation de notre projet de fin d'études. La gestion du temps sera alors primordiale, le logiciel Project 2016© de Microsoft™ est un excellent outil pour avoir le pouls réel de notre avancement dans le projet. Il nous incombera aussi de nous discipliner entre nous, les plaisanteries et les causeries sont

importantes pour notre santé mentale, toutefois elles ne font pas avancer notre projet, c'est pourquoi nous devons aussi planifier nos pauses.

N°	Description	Catégorie	Probabilité	Impact	Criticité	Mesure d'atténuation	Plan de réponse
1	Manque de temps	Méthode	Faible	Majeur	Significative	Bonne planification	Faire plus d'heures
2	Procrastination	Méthode	Extrême	Modéré	Élevée	Pauses planifiées	Temps supplémentaire

Figure 7 - Manque de temps : Table des risques

### 1.5.8 Quantité de portes logiques :

Le principe même d'un FPGA repose sur le nombre de portes logiques qui le composent. Dans notre cas, nous avons 100 000 portes logiques à notre disposition. Il est tout à fait justifié de se demander si la quantité est suffisante dans le cadre d'un projet de longue haleine comme celui-ci. Alors, il faut voir la question de la façon suivante : si on cherchait à compter le nombre de transistors impliqués dans un programme, il serait possible de le savoir, mais ce calcul serait très difficile à réaliser. Cependant, le FPGA a une force que les microcontrôleurs n'ont pas; nous pouvons compter le nombre de bits de signaux utilisés et le convertir en circuits à bascule. Par exemple, un bus parallèle avec 8 bits de données tel qu'illustré à la Figure 8 - Bus série parallèle.

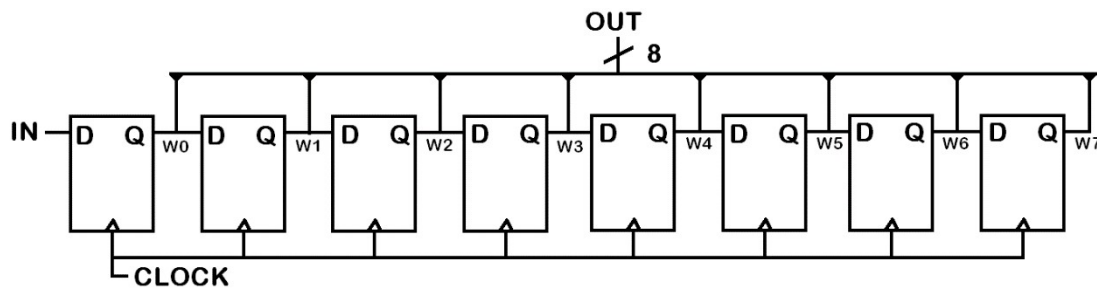


Figure 8 - Bus série parallèle

L'équipe précédente avait fourni à notre coach un schéma de leur projet final, qui nous a permis de compter le nombre de signaux nécessaires au projet. Nous arrivons à la somme estimée de 255 bascules D. Ce qui est amplement inférieur à la capacité de notre FPGA qui, selon les fiches techniques, contient 1920 bascules D. Le manque de bascules ne sera donc pas un problème dans notre cas, et si par malheur une telle situation advenait, nous aurons la possibilité de mettre un FPGA qui contient 9312 bascules D.

## 2 Liste des requis

#	Requis
1	Utilisation d'un FPGA (Modèle Xilinx XC3S100E)
2	Alimentation 5V principale
3	Alimentation 30V pour l'écran
4	Alimentation 1.2V pour le FPGA (VCCINT), 2.5V (VCCAUX) et 3.3V (VCCO)
5	Communication bidirectionnelle par le port parallèle, par I <sup>2</sup> C et RS-232
6	Communication bidirectionnelle par SPI (bonus)
7	Le port parallèle doit avoir 8 lignes de données, 6 lignes d'adresses et 5 chip selects
8	La communication I <sup>2</sup> C doit se faire à une vitesse de 100kHz
9	La communication SPI devrait se faire à une vitesse de 1MHz
10	La communication RS-232 pourrait supporter plusieurs taux de transferts différents: 19200 bauds, 38400 bauds, 57600 bauds et 115200 bauds.
11	Un connecteur USB serait utilisé pour le RS-232
12	Un circuit de conversion USB->Série devrait être présent (par exemple FT232R)
13	Une surface tactile capacitive devrait être sur l'écran
14	Le contrôleur renvoie les informations de la surface tactile au travers du protocole de communication employé.
15	Le FPGA a besoin d'une mémoire non-volatile EEPROM externe pour charger son programme
16	La matrice de caractère devra être programmée à même la logique du FPGA
17	Un connecteur 40-pins sera utilisé pour supporter la communication parallèle
18	Le projet devra tenir dans un boîtier
19	On tient à rendre le projet compatible avec la carte Dallas et la carte STM
20	On tient à rendre le projet compatible avec la carte PIC (bonus)
21	Les compatibilités du projet seront assurées grâce à de nombreuses méthodes de communication avec le contrôleur, qui seront supportées avec des bibliothèques fournies.
22	L'écran devra pouvoir afficher autant en mode graphique qu'en mode texte
23	L'écran devra pouvoir afficher en mode graphique et en mode texte en même temps
24	On devra pouvoir changer les modes via des commandes dans les modes de communication
25	Le projet consistera d'un seul PCB
26	La fréquence de rafraîchissement de l'écran est de 75Hz, et sera utilisée à sa pleine capacité.
27	Le PCB devra avoir un connecteur JTAG pour la programmation du FPGA
28	Le PCB devra avoir un bouton RESET pour réinitialiser l'affichage, les registres et la RAM

Figure 9 - Liste globale des requis

## 2.1 Liste des requis matériels

#	Ref	Requis
<b>M-1</b>	11	Un connecteur USB est utilisé pour la communication RS-232
<b>M-2</b>	17	Un connecteur 40-pin est utilisé pour supporter la communication parallèle
<b>M-3</b>	27	Le PCB doit avoir un connecteur JTAG pour la programmation du FPGA.
<b>M-4</b>	25	Le projet sera constitué d'un seul PCB, séparé en différentes parties : alimentation, contrôle, FPGA et communication.
<b>M-5</b>	13	Une surface tactile capacitive est sur l'écran
<b>M-6</b>	18	Le projet devra tenir dans un boîtier
<b>M-7</b>	1	Le projet sera basé sur un FPGA (Xilinx XSC3S100E)
<b>M-8</b>	12	Un circuit de conversion USB->RS-232 (FT232R)
<b>M-9</b>	15	Le FPGA nécessite une mémoire EEPROM externe
<b>M-10</b>	26	La fréquence de rafraîchissement de l'écran est de 75Hz
<b>M-11</b>	28	Bouton RESET pour réinitialiser l'affichage, les registres et la RAM

Figure 10 - Liste des requis matériels

## 2.2 Liste des requis électriques

#	Ref	Requis
<b>E-1</b>	2	Alimentation 5V principale
<b>E-2</b>	3	Alimentation 30V pour l'écran
<b>E-3</b>	4	Alimentation 1.2V pour le FPGA (VCCINT), 2.5V (VCCAUX) et 3.3V (VCCO)
<b>E-4</b>	5	Communication bidirectionnelle par le port parallèle, par I <sup>2</sup> C et RS-232
<b>E-5</b>	7	Le port parallèle doit avoir 8 lignes de données, 6 lignes d'adresses et 5 chip select
<b>E-6</b>	8	La communication I <sup>2</sup> C se ferait à une vitesse de 100kHz
<b>E-7</b>	10	La communication RS-232 pourrait supporter plusieurs taux de transferts différents, 19200 bauds, 38400 bauds, 57600 bauds et 115200 bauds.
<b>E-8</b>	6	Communication bidirectionnelle par SPI (bonus)
<b>E-9</b>	9	La communication SPI devrait se faire à une vitesse allant jusqu'à 1MHz

Figure 11 - Liste des requis électriques

### 2.3 Liste des requis logiciels

#	Ref	Requis
<b>L-1</b>	14	Le contrôleur renverrait les informations de la surface tactile au microcontrôleur au travers du protocole de communication en emploi.
<b>L-2</b>	16	La matrice de caractères sera programmée à même la logique du FPGA
<b>L-3</b>	19	Le projet devra être compatible avec l'ensemble Dallas et STM
<b>L-4</b>	21	Les compatibilités seront assurées grâce à de nombreuses méthodes de communication avec le contrôleur, qui seront supportées avec des bibliothèques fournies.
<b>L-5</b>	22	L'écran devra pouvoir afficher autant en mode graphique qu'en mode texte
<b>L-6</b>	23	L'écran devra pouvoir afficher en mode graphique et en mode texte en même temps
<b>L-7</b>	24	On devra pouvoir changer les modes via des commandes dans l'interface de communication
<b>L-10</b>	20	Le projet devra être compatible avec la carte PIC (bonus)

Figure 12 - Liste des requis logiciels

### 3 Architecture du projet

#### 3.1 Global – Niveau 0 :

##### Boîtier et écran; vue de l'extérieur

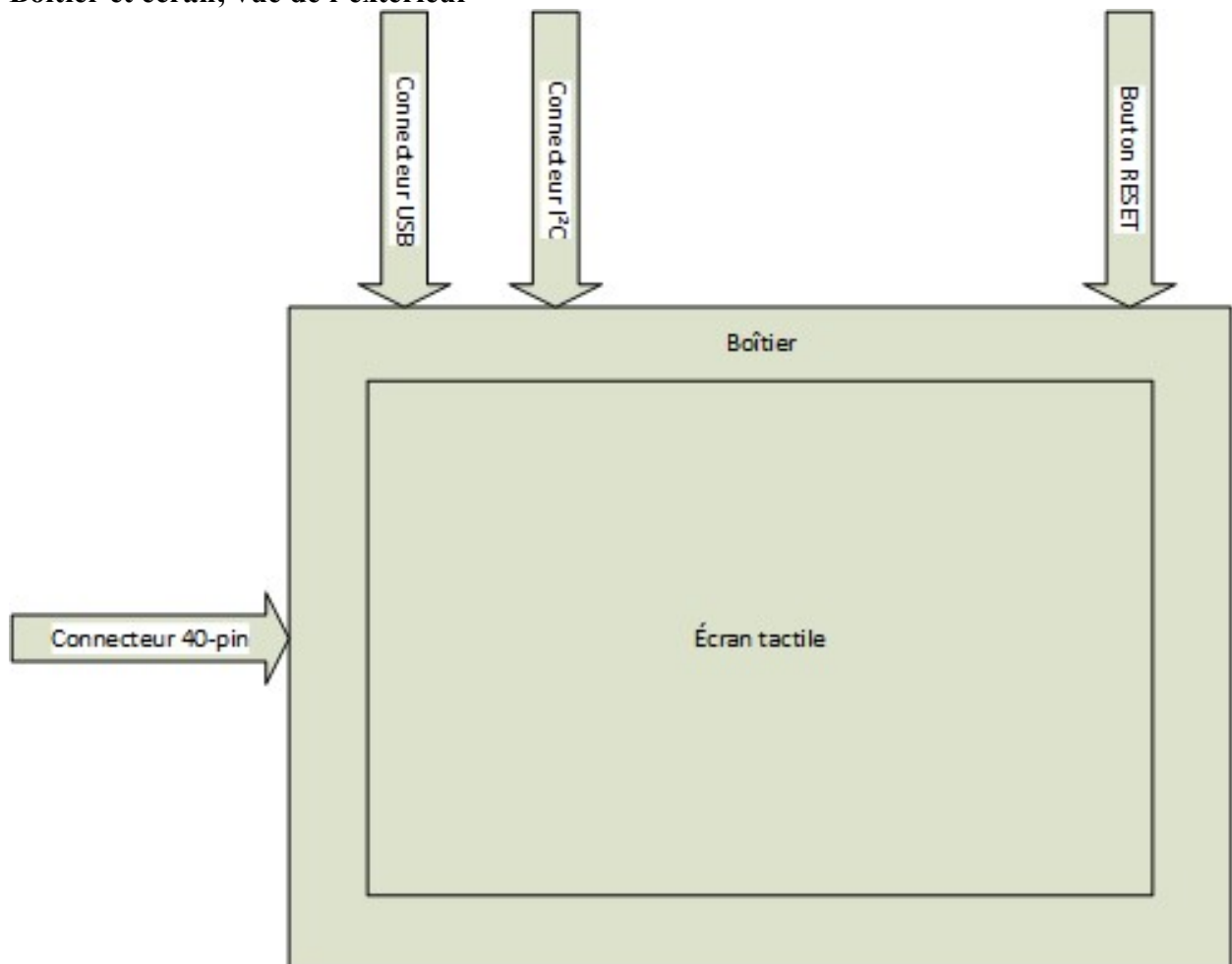


Figure 13 - Schéma Bloc Global (Niveau 0) : Boîtier et écran

Le contrôleur d'écran, vu de l'extérieur, sera contenu dans un boîtier imprimé par les imprimantes 3D du cégep. Il y aura dans le boîtier des espaces prévus pour un connecteur 40-broches pour la communication parallèle, un connecteur micro-USB pour la communication série RS-232 et un connecteur Molex 4-broches pour la communication de type I<sup>2</sup>C. Un bouton RESET sera également présent à l'arrière du boîtier pour réinitialiser le contrôleur. La pièce maîtresse du projet sera l'écran tactile, clairement visible sur le dessus du boîtier.



### 3.1.1 Global – Niveau I :

#### PCB, couches TOP et BOT

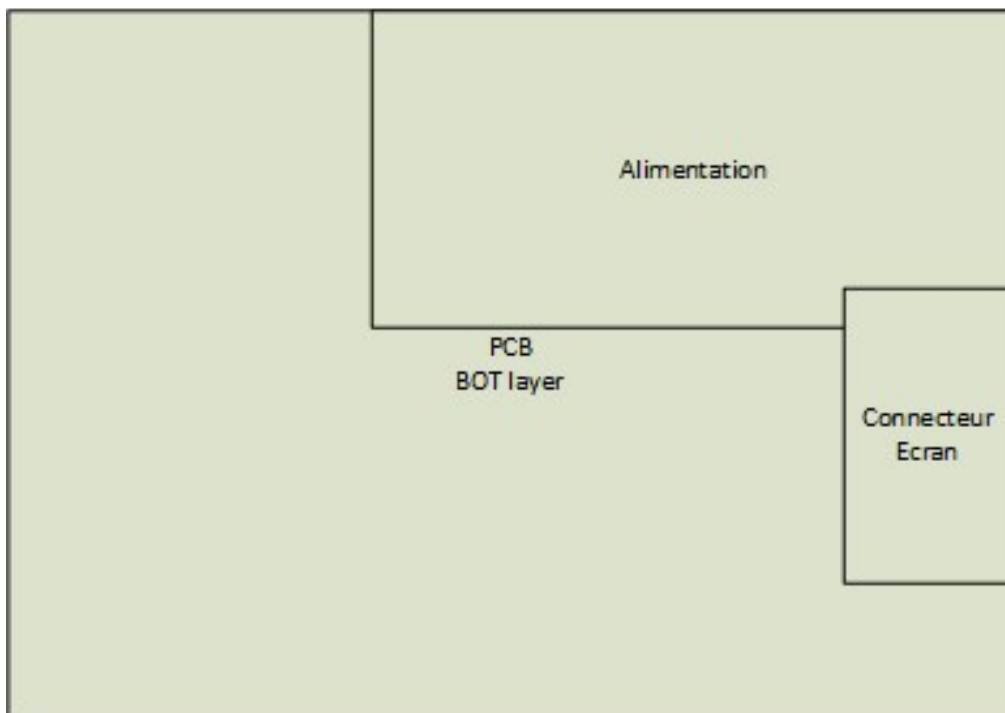
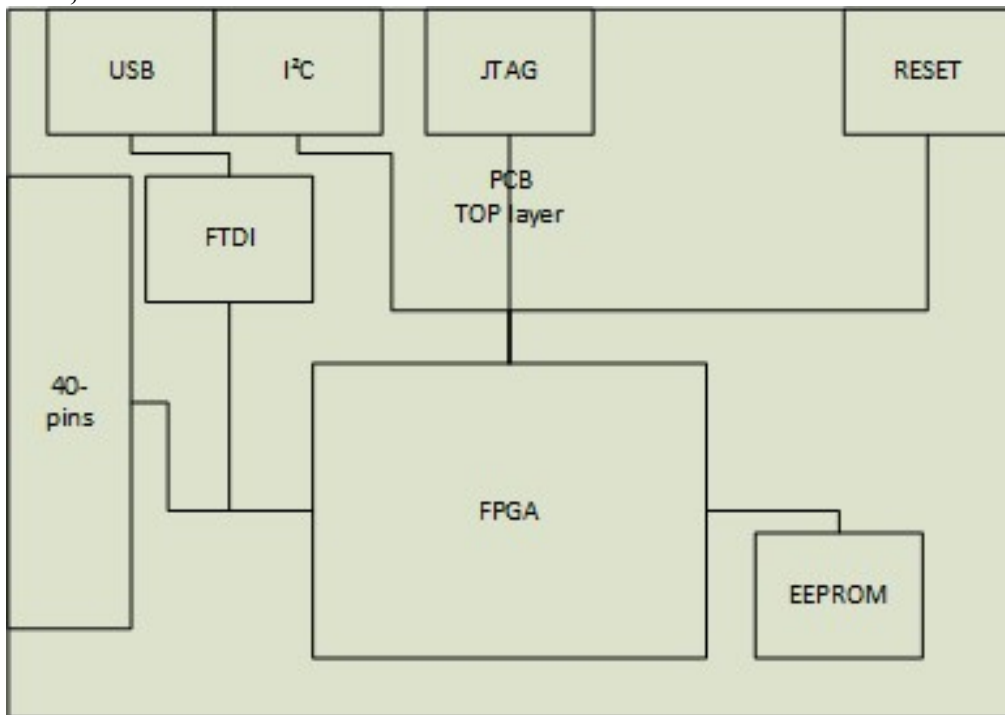


Figure 14 - Schéma bloc global (Niveau I) : PCB, couches TOP & BOT

Le projet est composé d'un unique PCB, qui doit être relativement petit, afin de respecter les contraintes de taille du boîtier. Le PCB comporte des entrées en haut et à gauche, des circuits de traitement situé dans le milieu et sa sortie vers l'écran situé à droite. L'alimentation se trouvera sur le bas du PCB, afin de laisser la place pour des bobines, des condensateurs et des dissipateurs de chaleur (car le PCB sera maintenu à distance du bas du boîtier par des supports). L'alimentation devra à être le plus possible distancé des composantes sensibles au bruit. Le connecteur JTAG est connecté directement au FPGA, tout comme le bouton RESET.

### 3.1.1.1 Alimentation – Niveau 2 :

#### Régulateurs de tension et alimentation à découpage

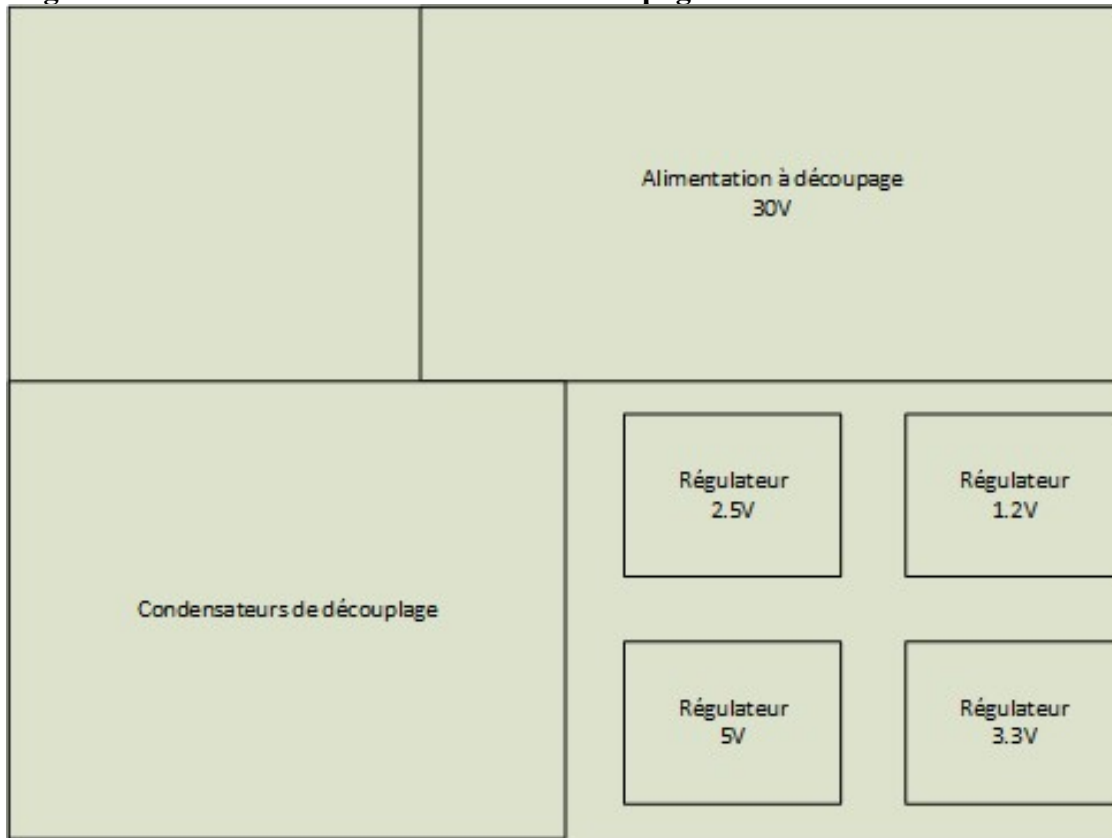


Figure 15 - Schéma bloc alimentation (Niveau 2) : Régulateurs de tension et alimentation à découpage

Le circuit nécessite de nombreux niveaux de tension : 25V, 5V, 3.3V, 2.5V et 1.2V respectivement. Ces niveaux de tensions devront être régulés et découplés. L'alimentation proviendra d'une source externe à 5V, qui sera descendue au niveau de tension appropriée par des régulateurs, et élevée à un niveau de 25V grâce à un circuit à découpage. De nombreux et gros condensateurs de découplage seront également nécessaires pour stabiliser les niveaux de tension.

### 3.1.1.2 Connecteur 40-broches – Niveau 2 : Connecteur 40-broches et communication parallèle

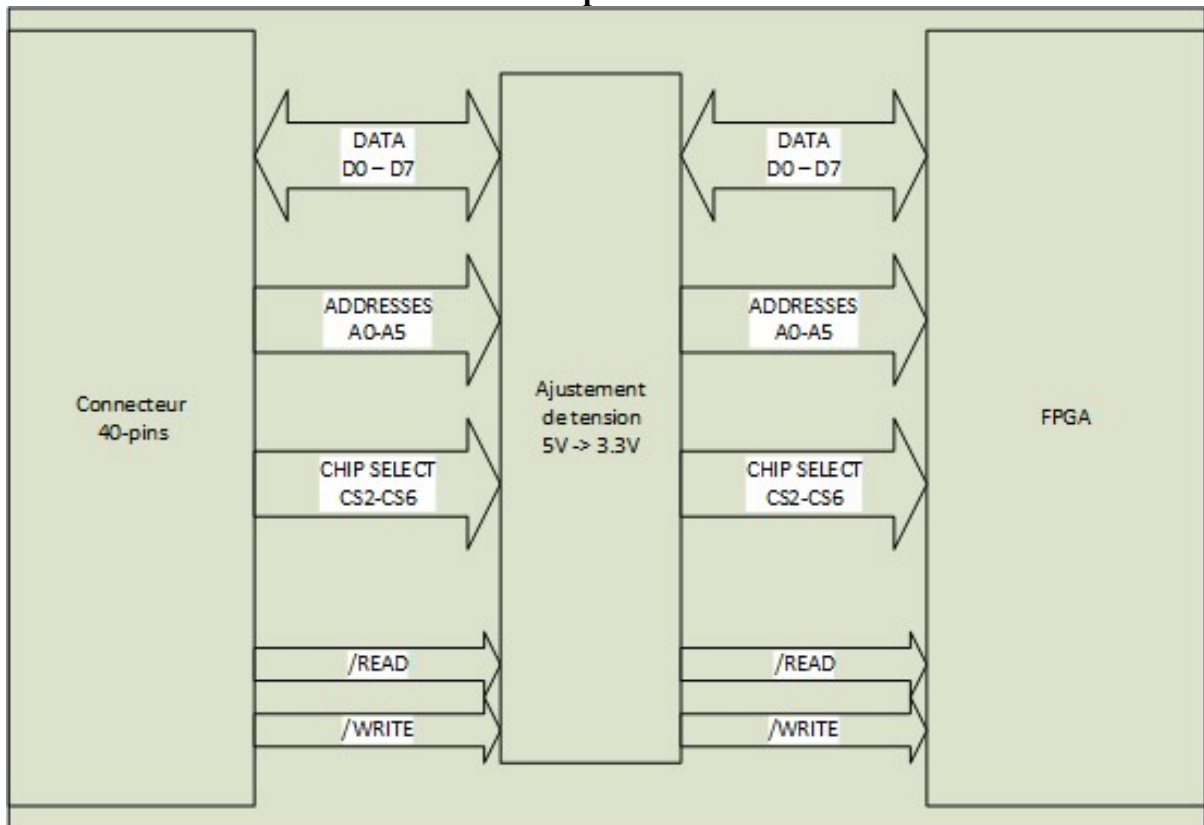


Figure 16 - Schéma bloc connecteur 40-broches (Niveau 2) : Connecteur 40-broches et communication parallèle

Le connecteur 40-broches, auquel sera connecté un câble IDE, permet de communiquer de façon parallèle avec le contrôleur. Cependant, il transporte ses informations sur des niveaux de tension 0V – 5V, et les entrées du FPGA sont à 3.3V. Il nous faudra donc convertir les niveaux de tension de 5V vers 3.3V pour les entrées. Des points de test seront également présents sur tous les signaux.

### 3.1.1.3 Connecteur USB – Niveau 2 :

#### Connecteur USB et communication UART

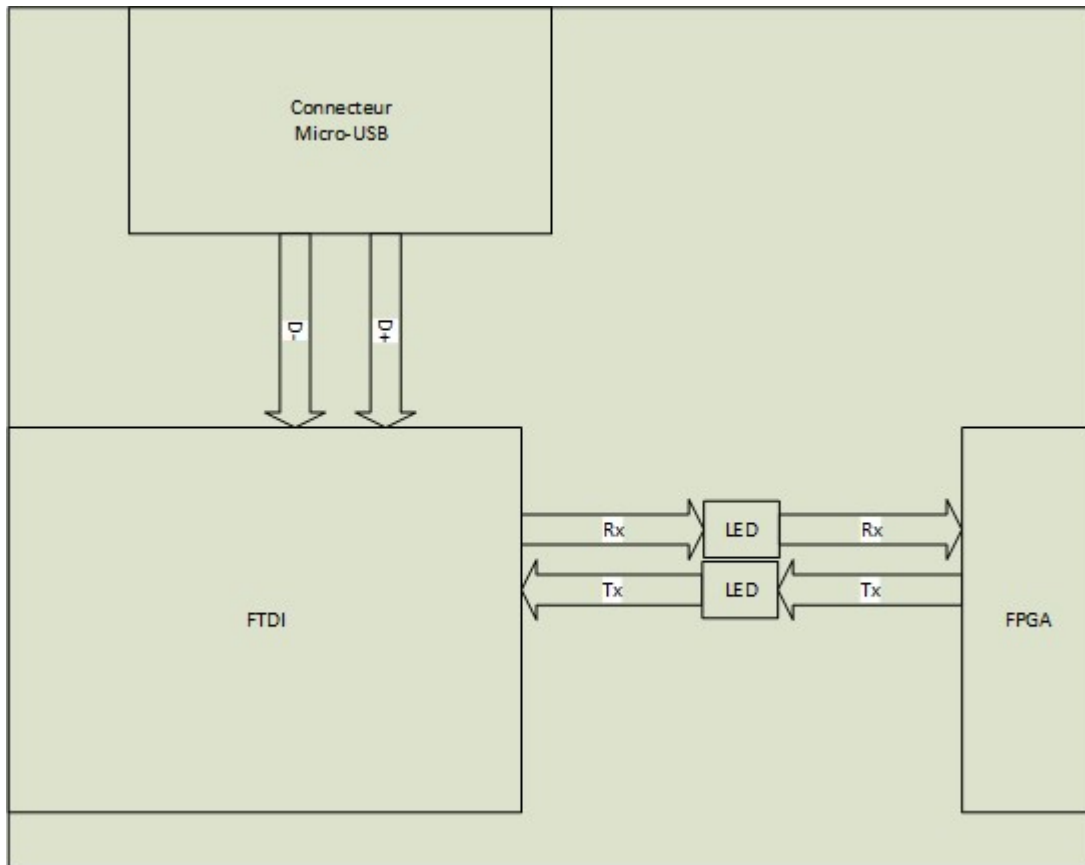


Figure 17 - Schéma bloc connecteur USB (Niveau 2) : Connecteur USB et communication UART

Le connecteur micro-USB permettra la communication UART vers le FPGA. On utilise un FTDI (modèle F232R) afin de convertir les informations de type USB vers des informations UART (sous la forme des signaux Rx et Tx) que le FPGA lira. Sur les lignes Rx et Tx se trouveront des LEDs pour signaler l'activité sur la ligne. Des points de test seront également présents afin de déboguer la carte plus facilement.

### 3.1.1.4 Connecteur I<sup>2</sup>C – Niveau 2

#### Connecteur Molex 4-broches et communication I<sup>2</sup>C

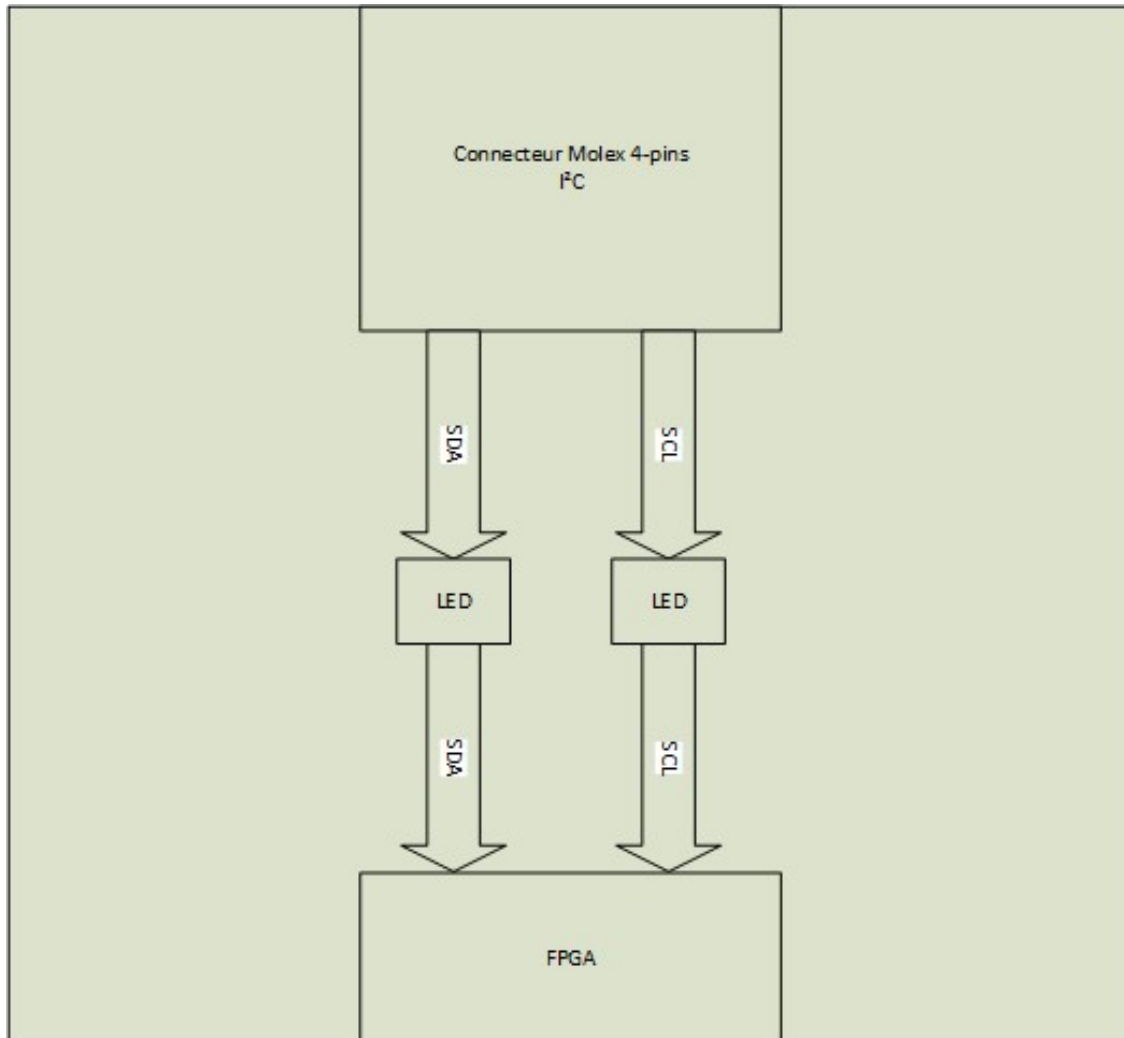


Figure 18 - Schéma bloc connecteur I<sup>2</sup>C (Niveau 2) : Connecteur Molex 4-broches et communication I<sup>2</sup>C

Le connecteur Molex 4-broches permettra la communication de type I<sup>2</sup>C. Des diodes électroluminescentes seront présentes pour indiquer une activité, et des points de test seront également disposés à des fins d'aide au déverminage.

### 3.1.1.5 Connecteur écran – Niveau 2

#### Connecteur 18-broches vers l'écran

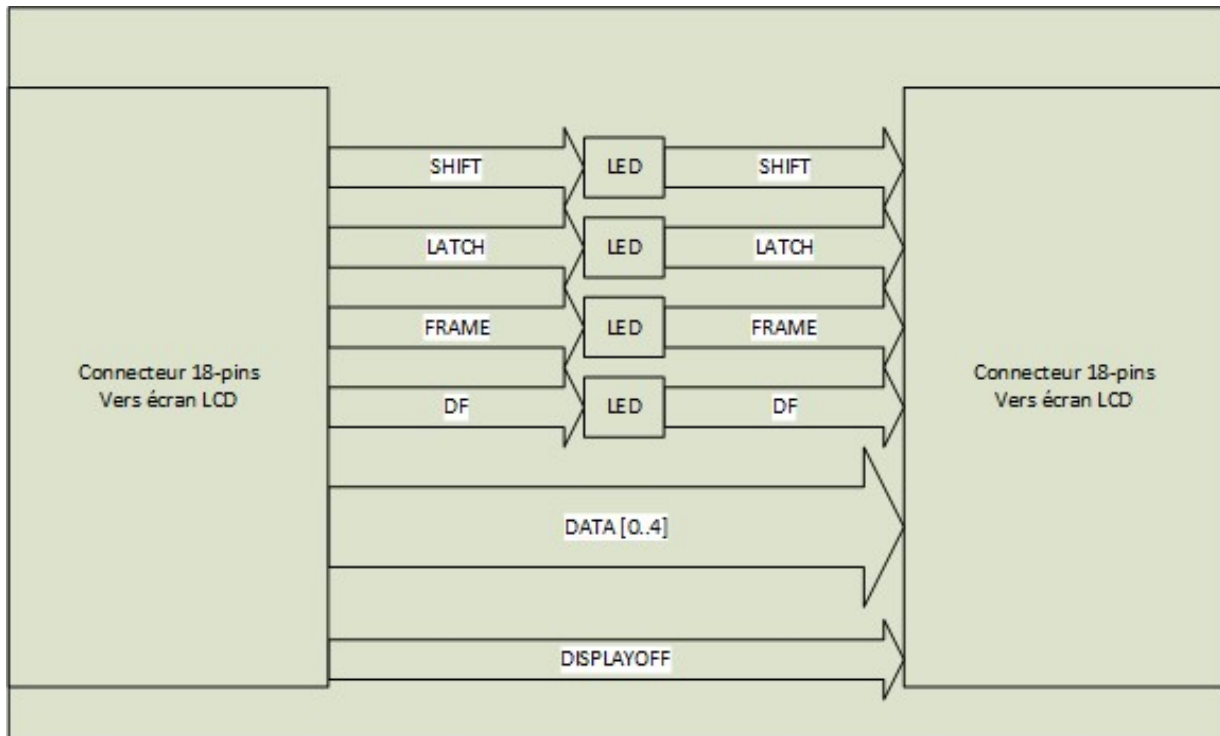


Figure 19 - Schéma bloc connecteur écran (Niveau 2) : Connecteur 18-broches vers l'écran

Le connecteur de l'écran dispose de 18 broches, dont 4 dédiées au bus de données 4-bits. L'information du bus de données est lue à chaque impulsion du signal SHIFT, également situé sur ce connecteur. À chaque 60 impulsions du signal SHIFT, un signal LATCH doit être envoyé pour changer de ligne. À chaque 160 impulsions du signal LATCH, un signal FRAME doit être envoyé pour indiquer une écriture complète, et l'état du signal DF doit être changé pour la même raison. Des diodes électroluminescentes seront présentes sur tous les signaux de timing, et des points de test seront disponibles sur tous les signaux du connecteur.

### 3.1.1.6 Dalle tactile – Niveau 2

#### Connexion entre la dalle tactile et le FPGA

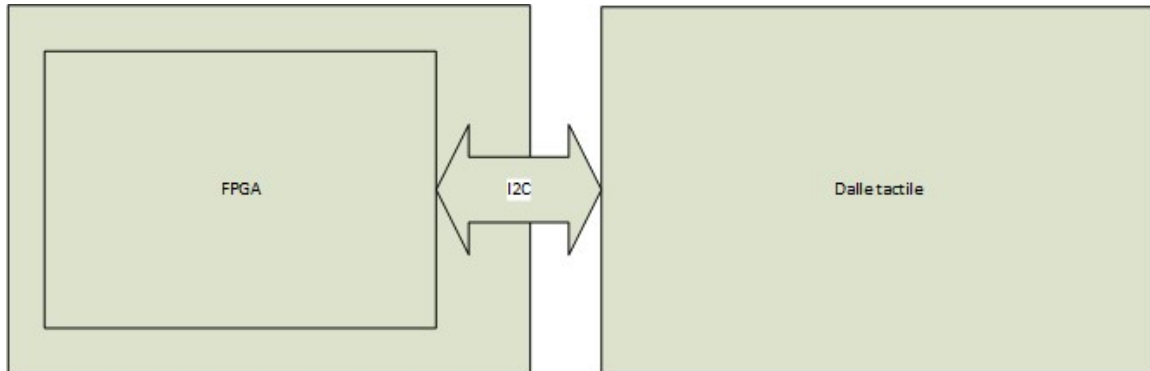


Figure 20 - Schéma bloc dalle tactile (Niveau 2) : Connexion entre la dalle tactile et le FPGA

La dalle tactile permettant de transformer notre simple écran en écran tactile vient avec un contrôleur (FT6236) déjà inclus. Cela nous permet d'employer cette surface via un lien de communication I2C.

### 3.1.1.7 Connecteur JTAG – Niveau 2

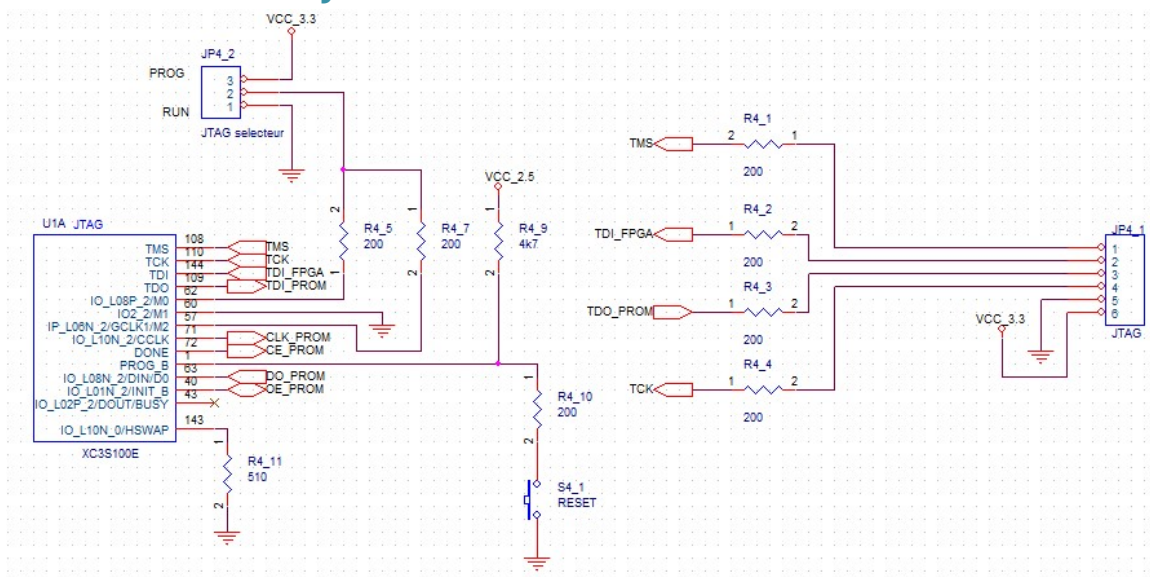


Figure 21 - Schéma Capture de la connexion du JTAG avec le FPGA

Le connecteur JTAG sera utilisé, dans le cadre du projet, à des fins d'exécution, de déverminage et de programmation. Il emploie le protocole standard JTAG pour interfacer le FPGA avec un ordinateur.

### 3.1.1.8 FPGA – Niveau 2

#### Entrées & Sorties du FPGA et EEPROM

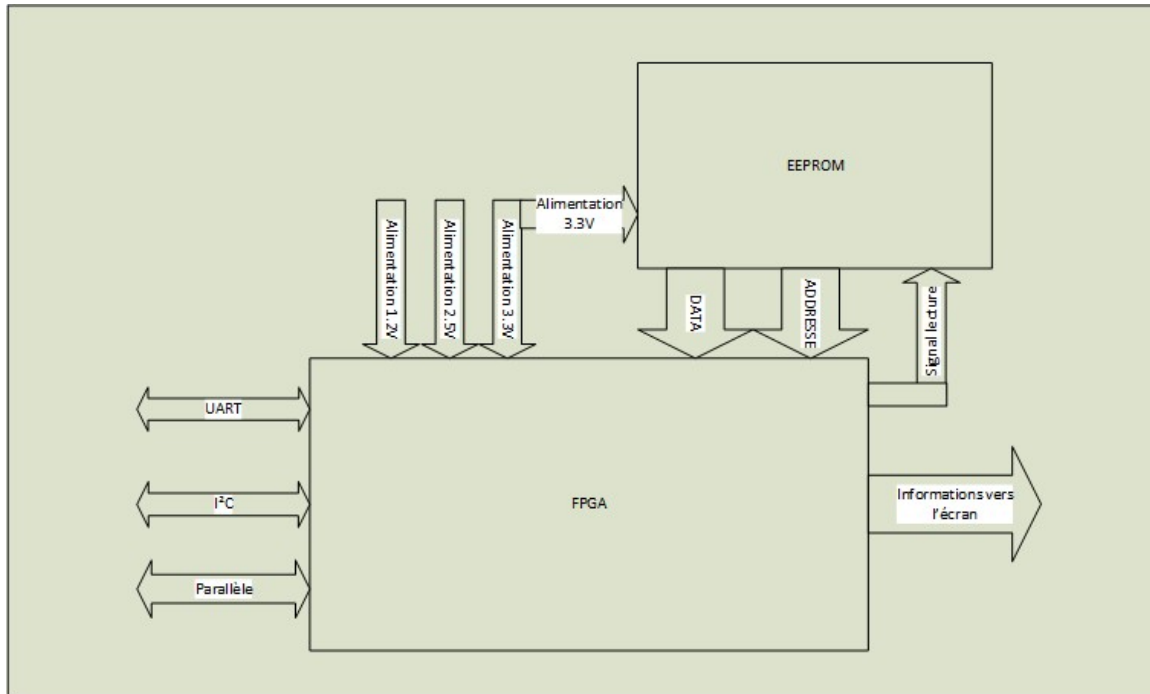


Figure 22 - Schéma bloc FPGA (Niveau 2) : Entrées & Sorties du FPGA et EEPROM

Le FPGA est la pièce maîtresse du projet. Elle prend les signaux de communication UART, I²C et Parallèle, interprète les informations et commandes, et les envoie vers l'écran au travers de son connecteur. Le FPGA a également besoin d'une mémoire morte de type EEPROM à l'externe pour charger le programme initialement, et de différents niveaux de tension (1.2V, 2.5V et 3.3V).



### 3.1.1.1.1 Alimentation à découpage – Niveau 3

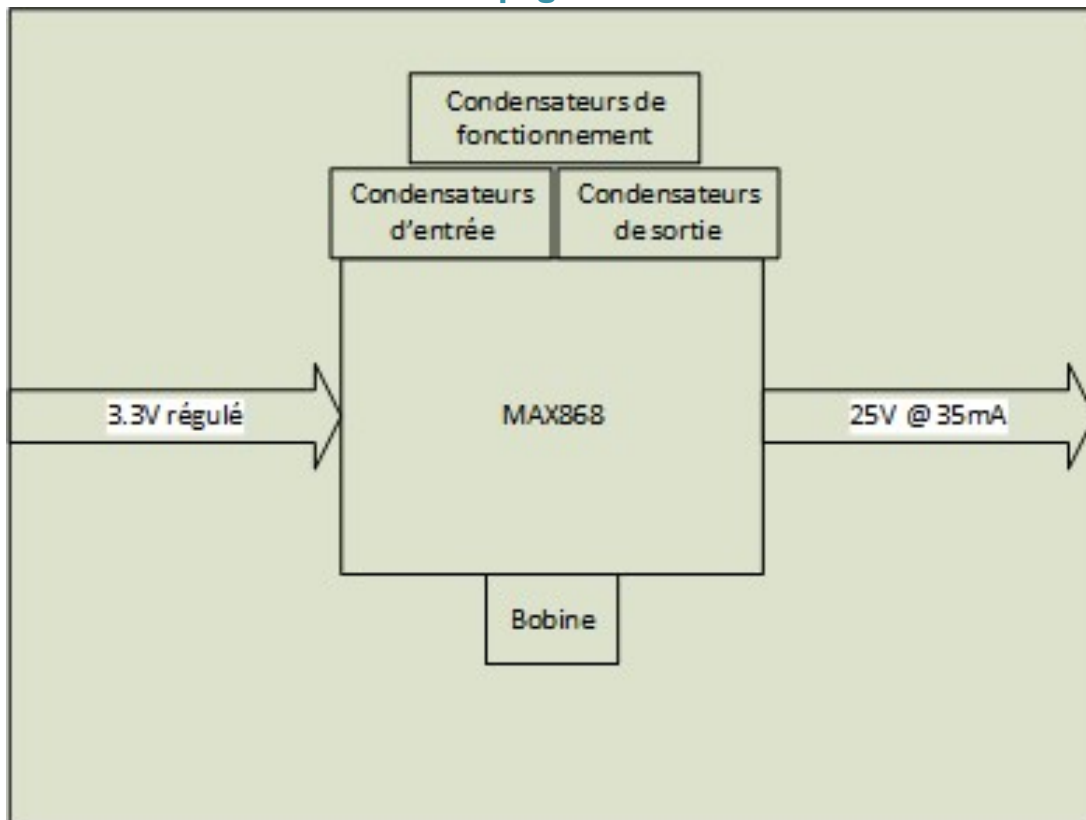


Figure 23 - Schéma bloc alimentation (Niveau 3)

Le circuit d'alimentation à découpage est nécessaire pour alimenter la « backplate » de l'écran. En utilisant une tension d'entrée à 3.3V, provenant d'un étage de régulateurs de tension, on utilise un circuit MAX686 et une bobine de 22 $\mu$ H afin de monter la tension à 25V. De nombreux condensateurs sont également nécessaires pour le circuit.

## 4 Planification du projet

### 4.1 Diagrammes WBS

#### 4.1.1 Documentation & Gestion

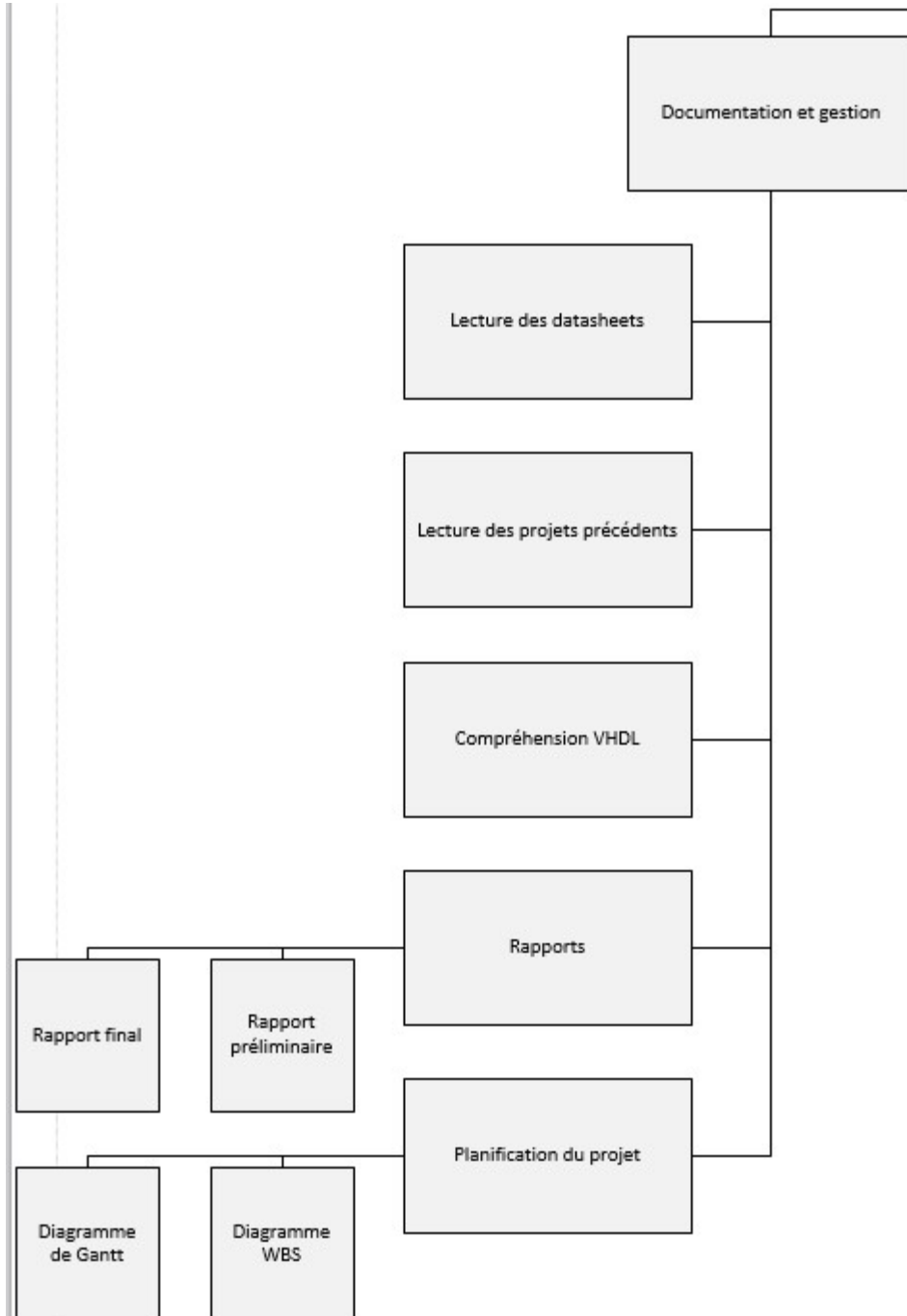


Figure 24 - WBS: Documentation & Gestion

#### 4.1.2 Alimentation

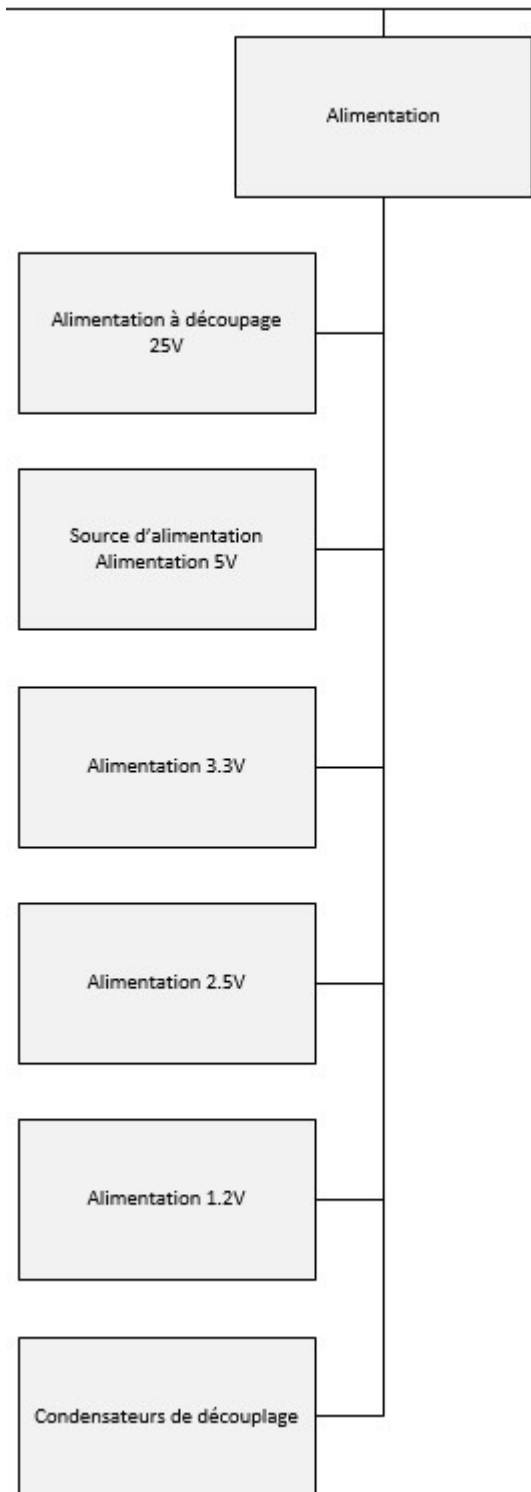


Figure 25 - WBS : Alimentation

#### 4.1.3 Communication

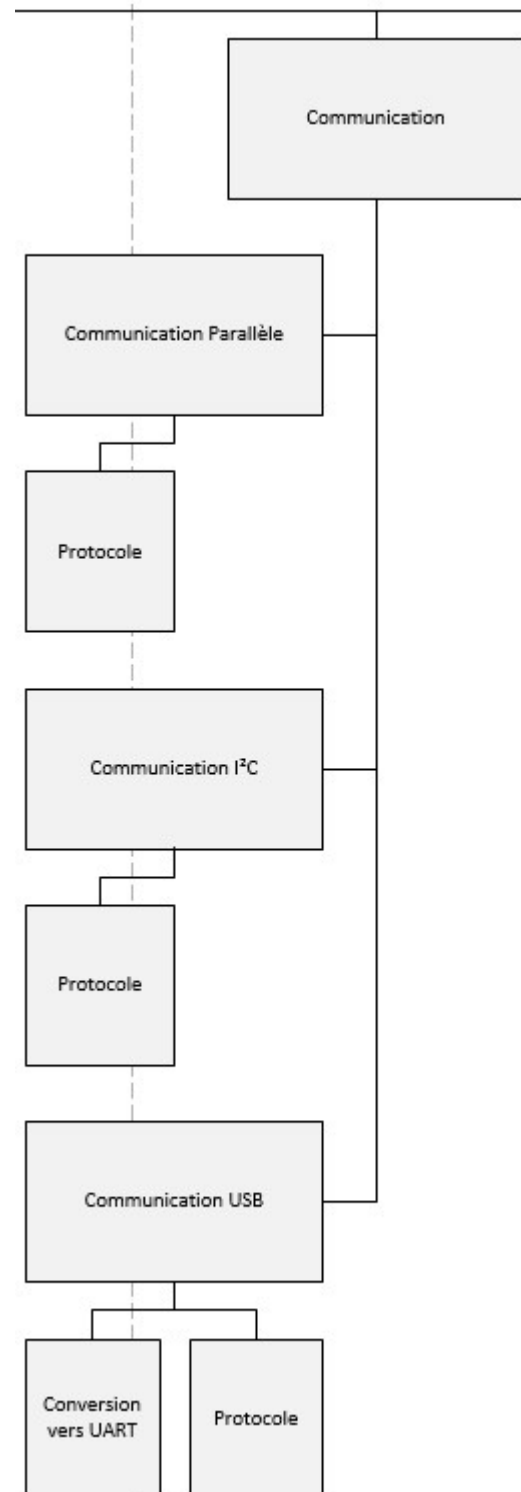


Figure 26 - WBS : Communication

#### 4.1.4 Traitement externe

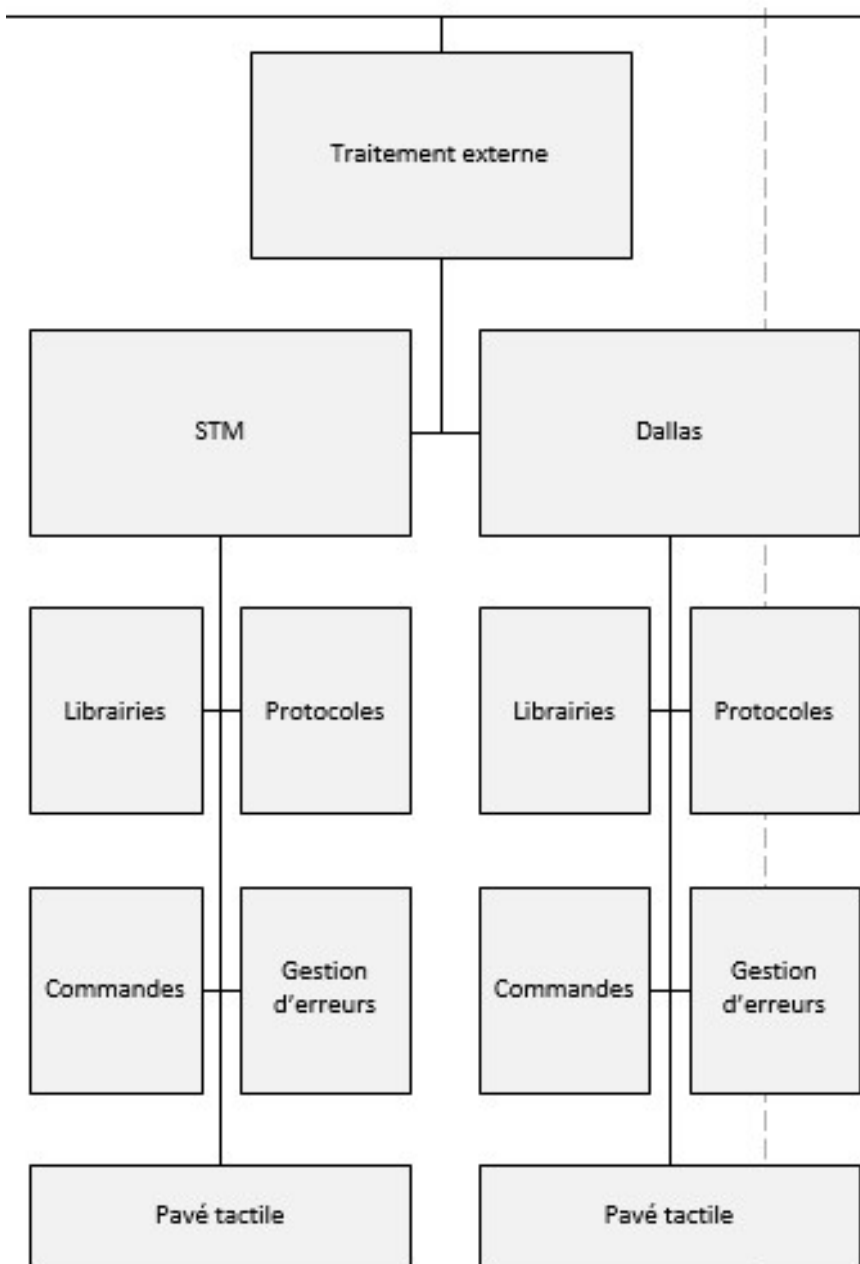


Figure 27 - WBS : Traitement externe

#### 4.1.5 Traitement interne

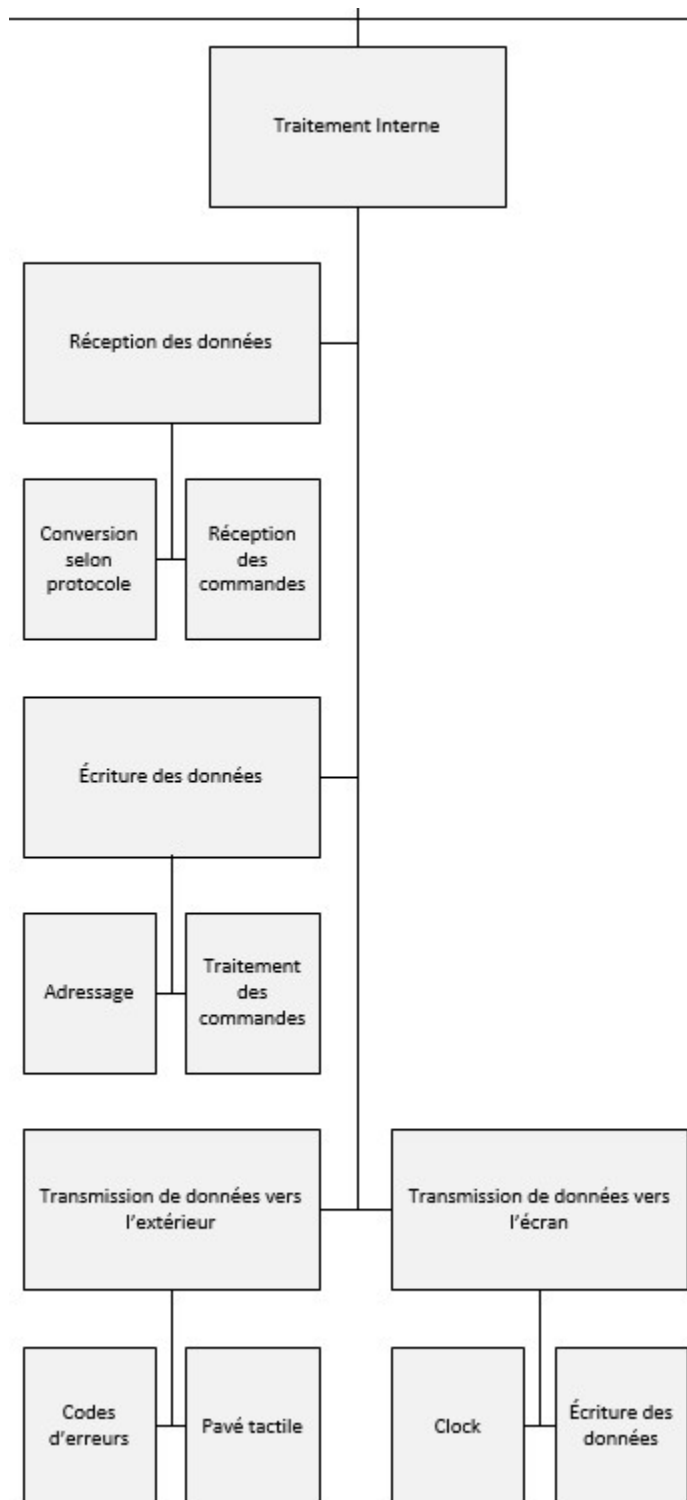


Figure 28 - WBS : Traitement interne

#### 4.1.6 Pavé tactile

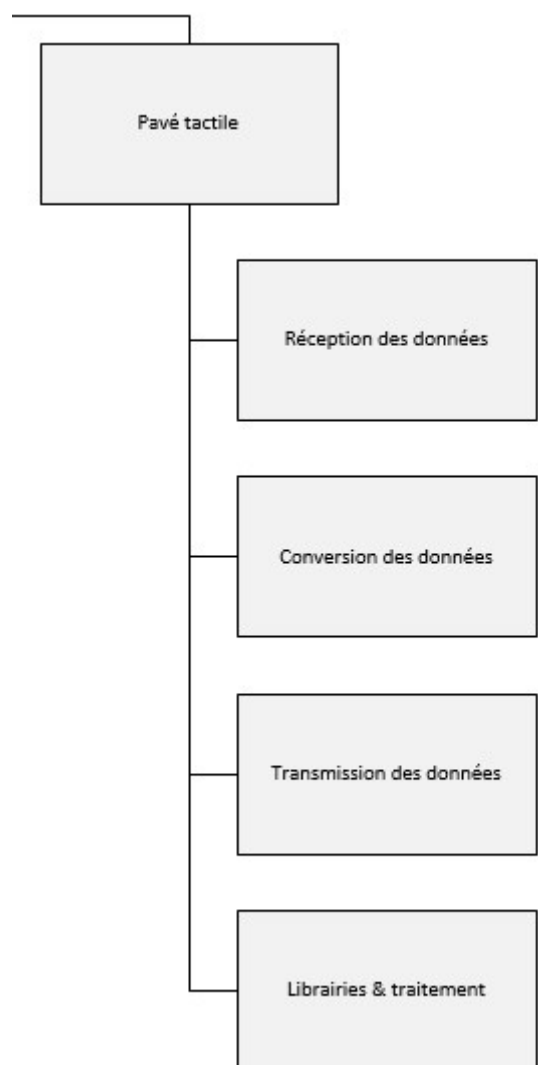


Figure 29 - WBS : Pavé tactile

4.1.7 Affichage

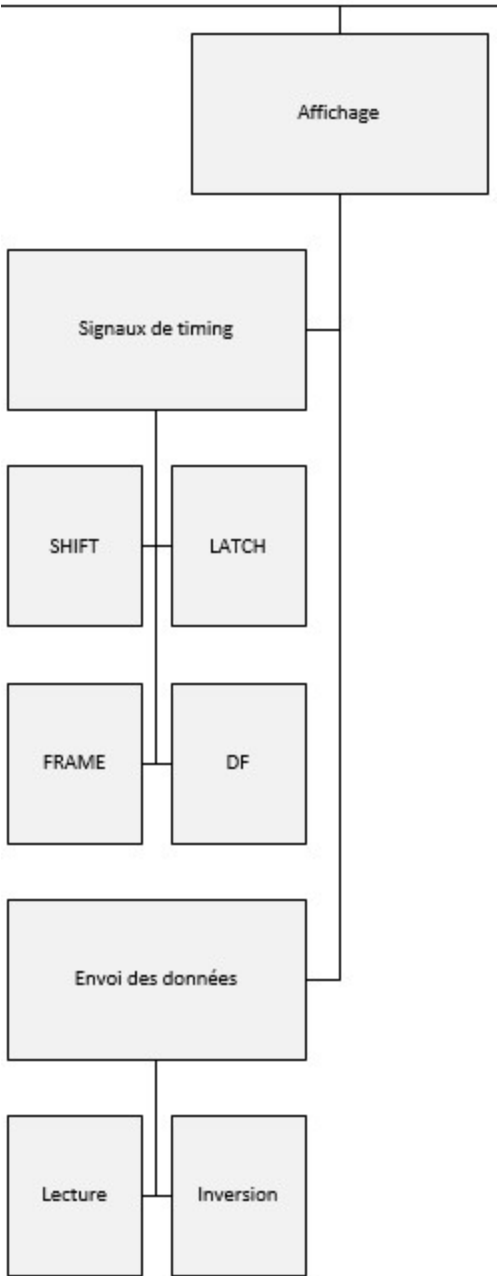


Figure 30 - WBS : Affichage

4.1.8 PCB & Boîtier

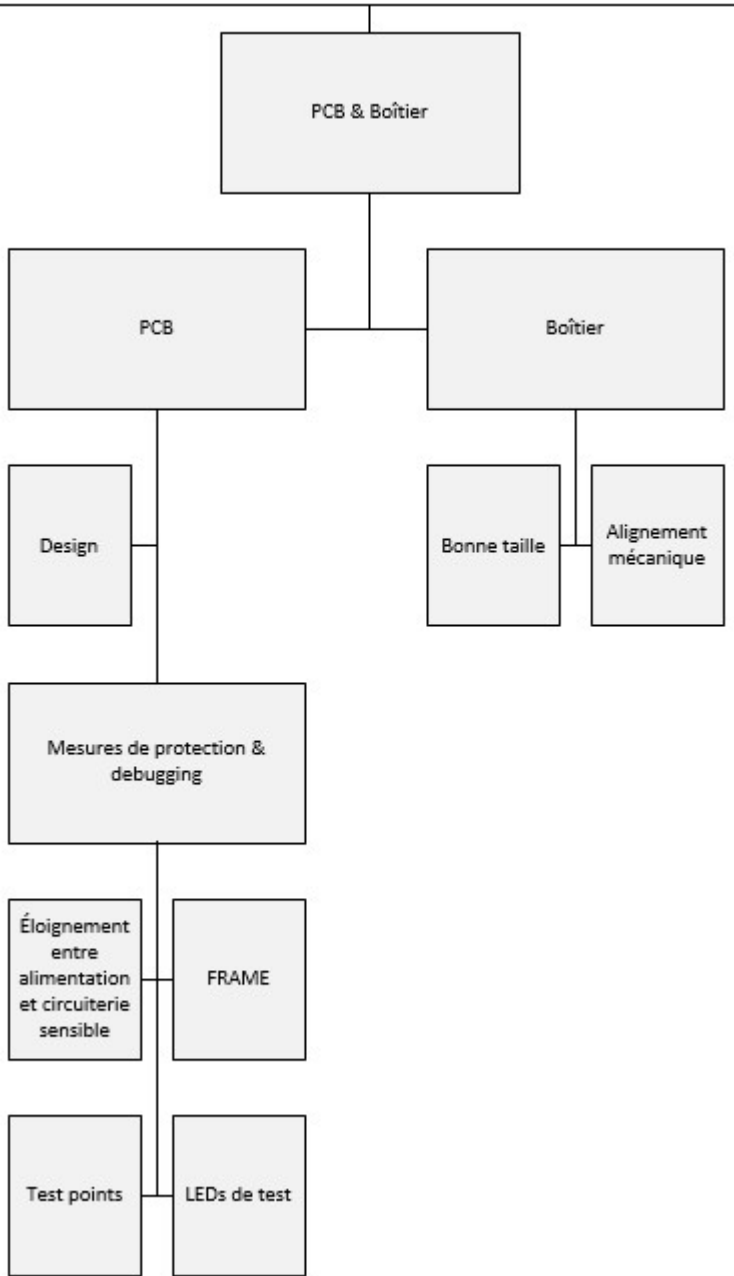


Figure 31 - WBS : PCB & Boîtier

## 4.2 Diagramme de Gantt

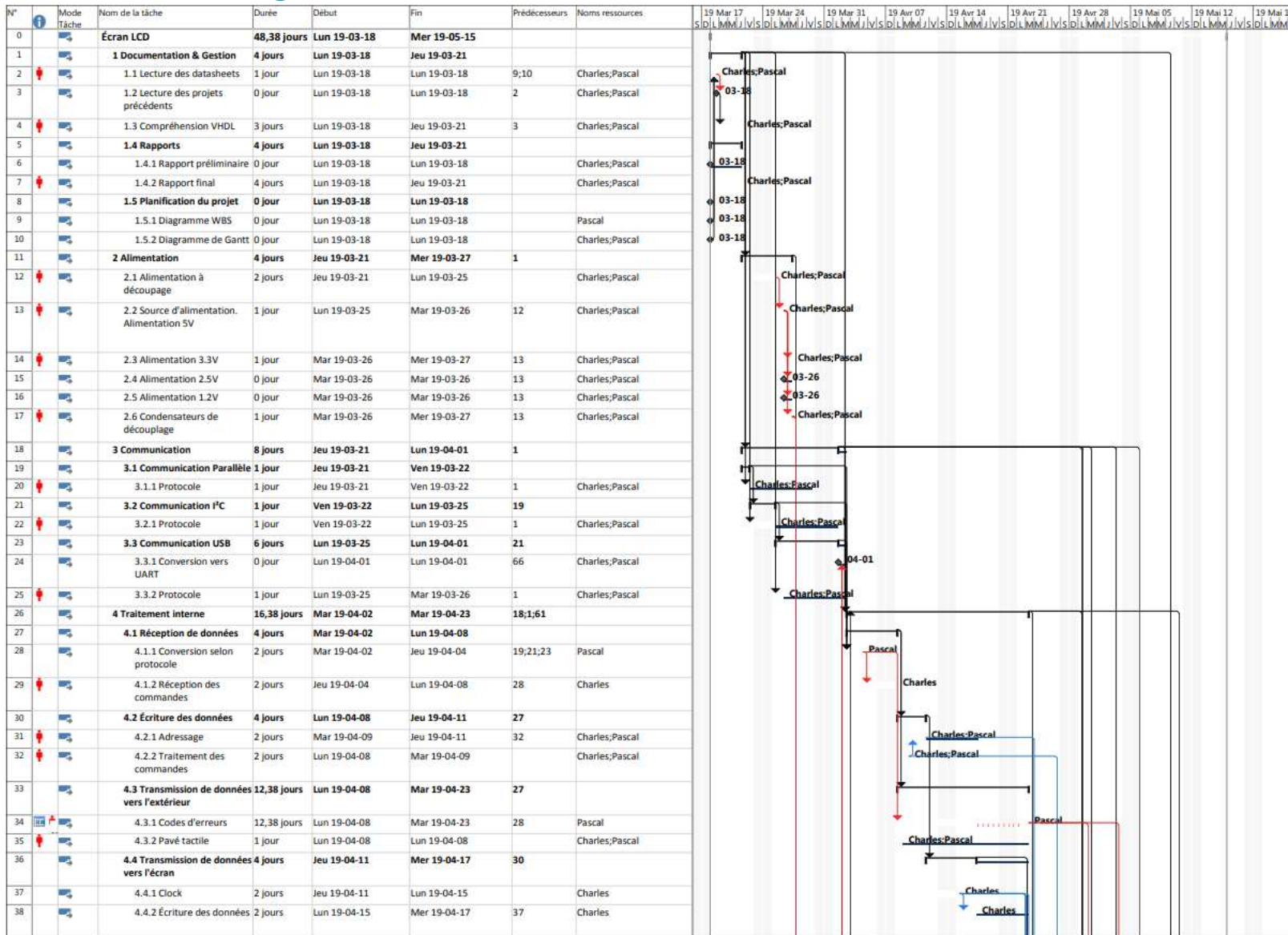


Figure 32 - Diagramme de Gantt - Partie I

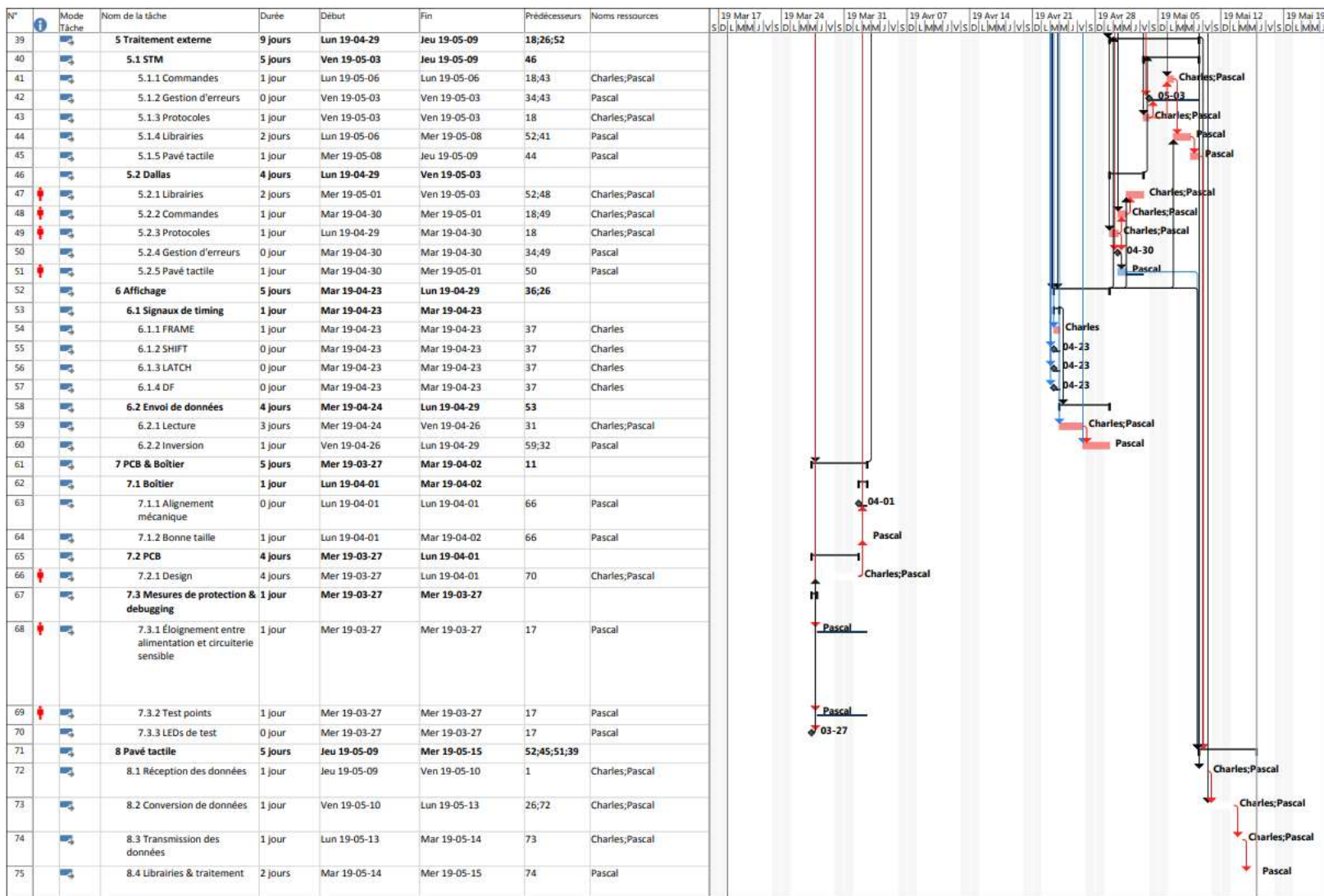


Figure 33 - Diagramme de Gantt - Partie 2