

Dossier technique

Un projecteur de logo

Hortense Allègre
Camille Diffiné
Etienne Jumpertz
Morgane Lardennois

Table des matières

Répartition du travail et planning	1
Team Électronique (Camille et Morgane)	1
Team Optique (Etienne et Hortense)	2
Cinquième et sixième séance : la mise en commun et la conception du prototype final	3
Rappel du cahier des charges	4
Schéma fonctionnel	4
Bloc de détection	5
Choix du montage électrique	5
Choix de la photodiode	5
Bloc de traitement	7
PIC16F1503	7
Description de l'algorithme utilisé	7
La sortie vers la source	9
Bloc d'émission	10
La source lumineuse	10
Les éléments du bloc optique	11
Boîtier d'alimentation	13
Conclusion	14
Notice d'utilisation	15

Répartition du travail et planning

Pour réaliser notre projet, nous avons séparé notre équipe en deux : Hortense et Etienne se sont occupés de la partie conception optique du projet, tandis que Camille et Morgane se sont occupés de la conception de la partie électronique et programmation du projet. Nous avons un planning global qui nous permettait de mettre des jalons réguliers pour juger de notre avancement ainsi que des plannings par équipe pour plus d'efficacité et pour pouvoir mieux s'adapter à la réalité du projet.

Team Électronique (Camille et Morgane)

Cette partie du projet comportait deux grandes parties : la conception du circuit électrique et la conception du programme informatique.

Dans un premier temps nous nous sommes concentrées sur la partie conception du circuit électrique, la programmation ne pouvant commencer que dans un second temps. Dans la première séance nous avons étudié séparément le circuit de détection et le circuit d'émission. Cette séance nous a notamment permis de réfléchir aux types de détecteurs que nous allions utiliser (des photodiodes) et comment nous pourrions avoir trois détecteurs différents dédiés chacun à une couleur (bleu, vert et rouge). Nous avons envisagé au début d'utiliser une tri photodiode mais les résultats n'étaient pas satisfaisants, la tri-photodiode étant trop peu sensible. Finalement nous avons opté pour 3 photodiodes classiques de type SFH206K recouvertes d'un filtre transparent coloré permettant de filtrer chaque couleur. Nous les avons branché en montage transimpédance afin d'augmenter leur sensibilité. En ce qui concerne la LED, nous attendions les conclusions de la team conception optique pour connaître la LED que nous allions utiliser, nous avons donc choisi de tester notre circuit à l'aide d'un bandeau de LED trichrome. Ceci nous a permis de concevoir un circuit qui fonctionne et dans lequel il nous suffirait juste de changer quelques résistances et transistors pour s'adapter à la LED finale. A la fin de cette séance, nous avons réussi à concevoir un système de photodétection et un système d'émission qui fonctionnaient séparément.

Lors de la deuxième séance nous avons essayé de regrouper les deux circuits en un en les reliant à l'aide d'un micro-contrôleur. Nous avons choisi le micro-contrôleur PIC16F1503 qui comportait suffisamment de sortie pour l'application que nous voulions réaliser. Nous avons dû prêter attention en regroupant les deux circuits au type de sortie auquel nous branchions les deux circuits car nous avons besoin de modules différents pour chacune des branches du circuit. Une fois la mise en commun des deux circuits bien entamée, nous avons pu commencer en parallèle la conception du programme. A la fin de la deuxième séance, les deux circuits étaient quasiment regroupés et la conception du programme tout juste commencée (initialisation des broches et idées de mise en oeuvre).

Lors de la troisième séance, nous avons quasiment fini de mettre en commun les deux circuits et nous avons bien avancé dans la programmation. Nous avons pu tester notre code sur le circuit. Au cours de la séance nous avons réussi à le faire fonctionner sur notre circuit. A ce moment-là nous étions en avance sur le planning que nous nous étions fixé en séance de pré-projet. Ainsi, nous avons donc décidé de rajouter une fonctionnalité au produit que nous allions proposer : un mode qui permet de choisir manuellement la couleur d'émission. Dans les dernières minutes

de la séance nous avons réfléchi à la mise en oeuvre de notre idée. Nous avons finalement décidé d'utiliser un bouton poussoir qui nous permettrait après un appui de faire défiler une roue de couleur et avec un deuxième appui de sélectionner la couleur sur la roue. Un troisième appui permettrait de repasser au mode automatique que nous avons réalisé précédemment. A la fin de cette séance, notre fonctionnalité "adaptation de la couleur" était opérationnelle.

Lors de la quatrième séance, nous avons tenté de mettre en place la nouvelle fonctionnalité et de la tester sur le circuit. Nous avons rencontré un certain nombre de problèmes et avons donc passé beaucoup de temps dans la séance à essayer de débogger le nouveau circuit électrique, ainsi que le code. Nous avons rajouté un bouton poussoir qui permet de choisir la fonctionnalité de notre projecteur. En parallèle, nous avons commencé à réaliser le circuit électronique sur le logiciel KiCad afin d'avoir le circuit imprimé pour les dernières séances. A la fin de cette séance la nouvelle fonctionnalité était bien avancée mais ne fonctionnait pas totalement et les schémas électroniques sur le logiciel KiCad étaient réalisés.

Lors de la cinquième séance, nous avons en parallèle fini de débogger le circuit et fini la conception du circuit imprimé dans un premier temps. Au bout de deux heures, nous étions en possession de notre circuit imprimé et le programme fonctionnait. Nous avons donc rejoint les autres pour la mise en commun des deux parties du projet.

Team Optique (Etienne et Hortense)

Cette partie avait pour objectif de réaliser la conception optique du système.

Dans un premier temps, il a fallu réfléchir à la conception du projecteur en lui-même. Pour cela, nous avons étudié au cours des premières séances un appareil à diapositives mis à notre disposition par le LEnsE.

Au cours de la première séance, nous avons découvert le fonctionnement de l'appareil à diapositives. Nous avons pu réaliser un schéma de l'appareil à diapositives en y notant les focales des lentilles que nous avons déterminées. Nous avons également passé beaucoup de temps à analyser à quelle distance de l'appareil, l'image serait la plus nette. A la fin de cette séance, nous savions comment fonctionnait un appareil à diapositives et nous avons les clés en main pour réaliser notre propre prototype.

Lors de la deuxième séance, nous avons démonté l'appareil à diapositives que nous avons étudié précédemment afin de récupérer les éléments qui nous intéressaient pour réaliser notre prototype. Nous avons donc récupéré les deux lentilles de focales respectives $f' = 30\text{mm}$ et $f' = 25\text{mm}$ avec la pièce métallique qui permet de les maintenir ensemble à la bonne distance l'une de l'autre. Nous avons également cherché quel objectif à utiliser après les deux lentilles. Nous avons fait différents essais avec des objectifs de focale 120 mm, 200 mm et 80 mm. L'objectif qui semblait le mieux adapté était celui de 120 mm. En effet, l'image était nette sur une grande plage en fonction de la distance entre l'objectif et la dernière lentille. A la fin de cette séance, nous avons réalisé la partie optique théorique : nous savions exactement quel montage utiliser.

Pendant la troisième séance, nous nous sommes penchés sur la source lumineuse à utiliser. En effet, il nous fallait une source lumineuse très puissante mais qui ne chauffe pas trop et surtout qui puisse éclairer en rouge, vert et bleu. Nous avons fait différents tests avec quelques LEDs

disponibles dans le LEnsE. Nous avons commencé avec des LED blanches Luxeon pour déjà assimiler comment elles fonctionnaient : montage sur un radiateur, différents branchements... Nous avons ensuite utilisé une LED trichrome mais nous avons rencontré des difficultés à ce moment-là : on voyait sur l'image obtenue en sortie du système optique des points distincts de couleurs qui correspondaient aux petites LED de couleur constituant la LED trichrome. Nous avons donc envisagé d'utiliser une opaline et avons fait des tests en conséquence. Avec l'opaline, la puissance était beaucoup trop diminuée et on ne pouvait obtenir une image que très proche du projecteur (environ 1 m). Cette séance nous a permis de nous familiariser avec les LEDs à utiliser et de comprendre pourquoi on avait besoin d'une LED de puissance trichrome.

Au cours de la quatrième séance, nous avons cherché quelle LED nous pourrions utiliser pour notre projecteur. Il n'y avait pas ce que nous recherchions au LEnsE donc nous avons passé un bon moment à faire des recherches sur internet pour trouver la LED la plus adaptée. Nous avons fini par trouver une LED trichrome avec dôme. La puissance était inférieure à ce que nous recherchions au départ mais nous n'avons pas trouvé mieux. D'autre part, le dôme permettait d'avoir un éclairage uniforme, dans lequel on ne distinguerait pas les différents points lumineux. Nous avons donc commandé cette LED chez LED Engin (référence LZP-00MD00). Après avoir effectué cette commande, nous avons rejoint la team électronique pour commencer à appréhender la jonction entre les deux.

Cinquième et sixième séance : la mise en commun et la conception du prototype final

Lors de la fin de la cinquième séance, nous avons commencé les soudures et la mise en place de notre produit final. Pendant que Camille, Etienne et Hortense travaillaient sur le circuit imprimé, les soudures et le choix de la protection des LEDs très puissante que nous allions utiliser dans le prototype final, Morgane a recherché une solution pour alimenter le dispositif.

En ce qui concerne la protection des LEDs, nous avons finalement choisi de prendre des résistances en puissance pour protéger notre LED par manque de temps pour trouver une solution moins gourmande en énergie. Ceci constitue une des améliorations possibles de notre dispositif.

Pour l'alimentation nous avons choisi d'utiliser le convertisseur alternatif-continu FKH2- 50-04 pour transformer le courant du secteur en courant continu et ainsi pouvoir alimenter nos deux alimentations : l'alimentation 466 6878 (plus ou moins 15V, 1,5-2,4 A) qui nous permet d'alimenter le circuit et l'alimentation 621 0685 (plus ou moins 24 V , 6,5 A) qui alimente la LED.

Rappel du cahier des charges

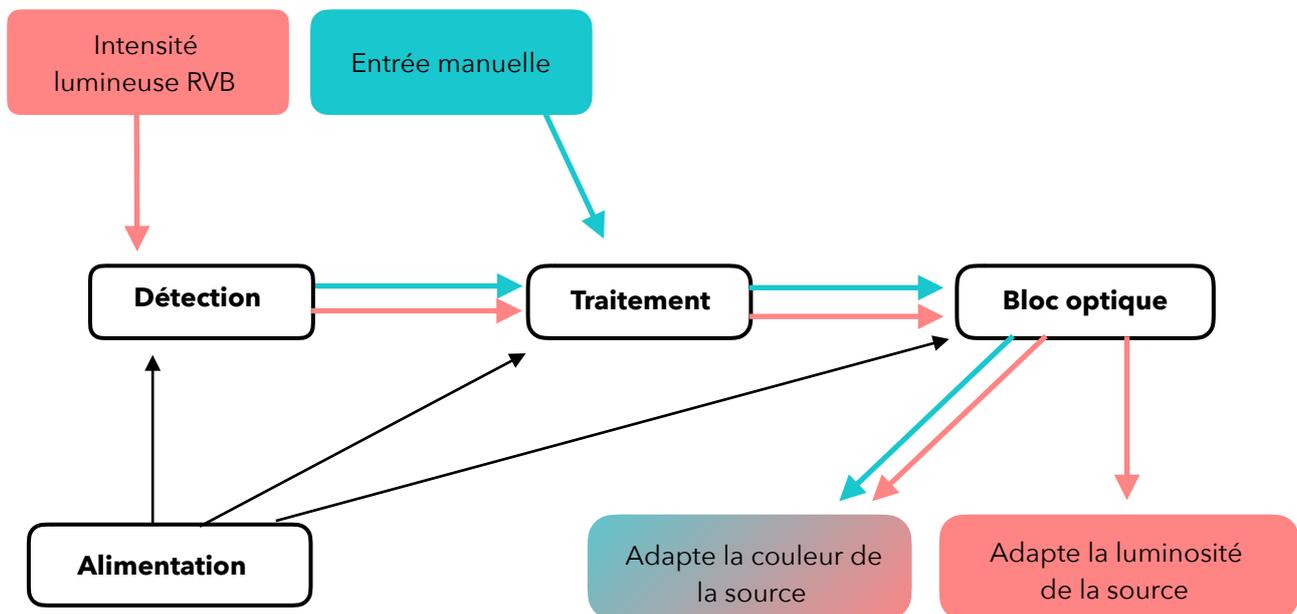
Notre projet vise à concevoir un projecteur de logo qui s'adapte à la lumière ambiante (en couleurs et en luminosité).

Le cahier des charges comprend les éléments suivants :

- Adaptation de la couleur du logo selon la couleur dominante de la pièce
- Adaptation de la luminosité de la source selon celle de la salle
- Projection à au moins 1m
- Faible encombrement
- Alimentation par le secteur.

Schéma fonctionnel

On peut décomposer le projet de la façon suivante :



Bloc de détection

L'entrée de notre système est composée de trois photodiodes dans le domaine du visible, de types SFH 206, munies chacune d'un filtre transparent afin de sélectionner une plage spectrale. Le but est de réaliser un capteur RGB qui fournit un courant proportionnel à l'intensité de chacune des composantes rouge, verte et bleue de la lumière ambiante.

Choix du montage électrique

On choisit de réaliser trois montages (un pour chaque couleur) de photodétection à l'aide d'un montage transimpédance.

Pour les trois montages transimpédance, nous avons choisi d'utiliser l'amplificateur opérationnel TL084, car il comprend directement quatre AOP (nous en avons besoin de 3).

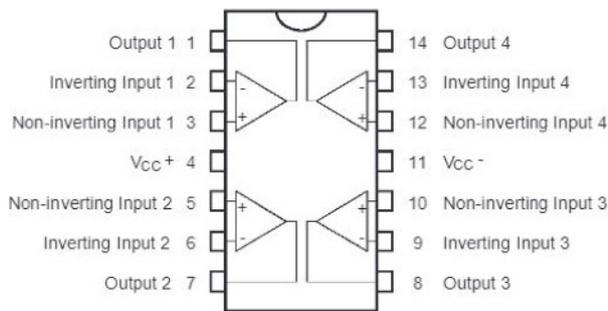


FIGURE 1 - Schéma du TL084 (issu de la documentation technique)

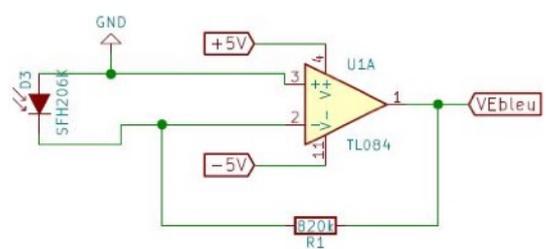


FIGURE 2 - Montage en transimpédance.

Nous avons fait le choix du montage en transimpédance afin d'augmenter la sensibilité du détecteur et les performances de notre circuit. Il nous reste à choisir la valeur de la résistance en fonction de la photodiode utilisée pour optimiser notre montage de photodétection.

Choix de la photodiode

Pour concevoir un capteur RGB, il faut que l'on associe une photodiode à chaque couleur que l'on veut détecter (rouge, vert et bleu). Nous avons dans un premier temps envisagé d'utiliser une tri-photodiode de type KPS-5130PD7C. Cependant, ces photodiodes n'étaient pas assez sensibles : la largeur spectrale de chaque couleur était trop importante (chevauchement de spectre entre les sensibilités des différentes photodiodes) et la sensibilité au flux de photons trop faible (sûrement à cause d'une surface réceptrice trop petite) pour détecter des variations de la luminosité ambiante. Nous avons donc dû changer moyen de détection photodiode.

Nous avons donc décidé d'utiliser trois fois la même photodiode, sensible dans le visible, la SFH206K, munies de filtre coloré pour sélectionner des plages spectrales.

On constate que l'on ne peut pas avoir la même sensibilité pour les trois couleurs du fait des caractéristiques de la diode mais aussi à cause des filtres utilisés. La photodiode seule est 5 fois plus sensible dans le rouge que dans le bleu et 2 fois plus sensible dans le rouge que dans le vert. Pour résoudre ce problème, nous allons choisir intelligemment nos résistances pour compenser la sensibilité de la photodiode. En prenant en compte simultanément la sensibilité de la photodiode et la transmission des filtres utilisés, on choisit de prendre R_{bleu} plus grande que R_{rouge} et $R_{\text{verte}} = R_{\text{rouge}}$, les problèmes de transmission des filtres rouge et vert compensant quasiment les problèmes de sensibilité de la photodiode.

C'est pourquoi finalement nous avons : $R_{\text{rouge}} = R_{\text{verte}} = 330 \text{ k}\Omega$ et $R_{\text{bleu}} = 820 \text{ k}\Omega$.

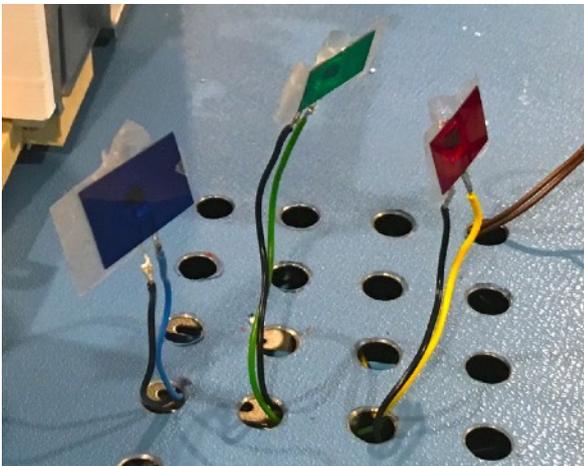


FIGURE 3 - Les trois photodiodes munies de leurs filtres de couleur.

Relative Spectral Sensitivity ^{1) page 8}

$$S_{\text{rel}} = f(\lambda)$$

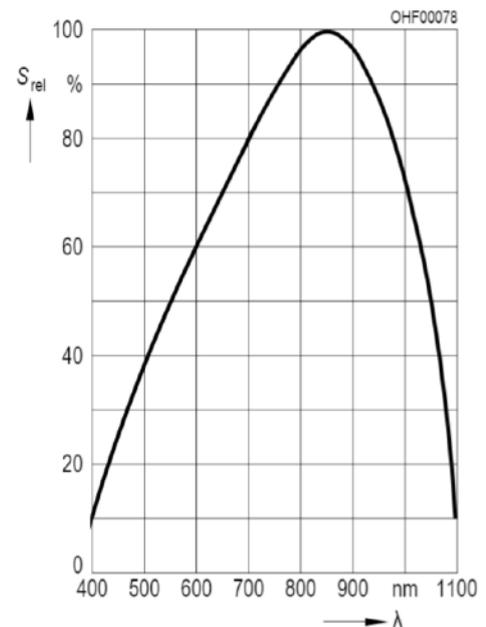


FIGURE 4 - Sensibilité de la SFH206K (graphe issu de la documentation technique)

Bloc de traitement

PIC16F1503

Le traitement des trois entrées venant des photodiodes est assuré par un microcontrôleur PIC16F1503, qui se charge de leur conversion analogique-numérique, ainsi que la génération d'une sortie modulée en conséquence.

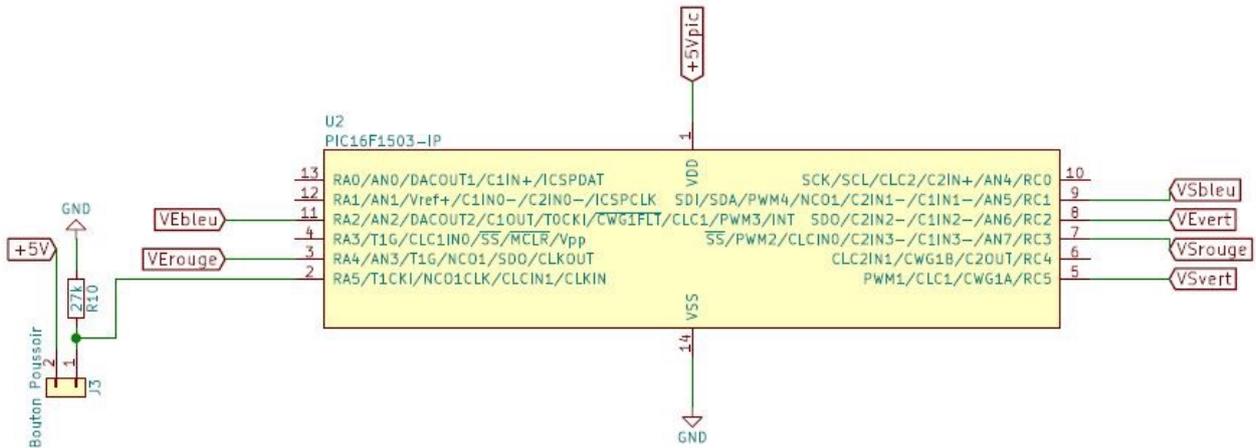


FIGURE 5 - Montage du PIC16F1503.

L'entrée 1 du pic est son alimentation, que l'on alimente à 5V, tandis que l'entrée 14 est la masse.

A l'entrée 2, nous avons l'arrivée du bouton poussoir afin de sélectionner le mode que l'on souhaite.

Les entrées 3, 8 et 11 sont branchés à la sortie des trois photodiodes.

Les sorties PWM 5, 7 et 9 sont branchés aux trois canaux correspondant aux couleurs de la LED.

Description de l'algorithme utilisé

Notre programme comporte 4 unités : un convertisseur analogique-numérique permettant de convertir l'information provenant des photodiodes, un module PWM pour pouvoir avoir un contrôle en puissance des LEDs du projecteur, un mode d'interruption qui permet de sélectionner un mode de fonctionnement du programme, et le corps du code.

Convertisseur analogique numérique :

La conversion analogique numérique et notamment possible à l'aide de la fonction *initADC* dans laquelle on initialise les entrées-sorties qui vont nous servir dans la conversion analogique

numérique ainsi que l'horloge de la conversion. Nous n'avons pas de critère particulier sur le temps de réponse de notre circuit c'est pourquoi nous avons choisi une horloge à 2MHz.

Module PWM :

Le mode PWM (*Pulse Width Modulation*) est un mode qui permet de moduler un signal en largeur d'impulsions. Pour pouvoir contrôler la signal modulé en largeur d'impulsion qu'envoie le PIC, il faut écrire une fonction qui permet de changer le rapport cyclique du signal modulé. Dans notre cas, les 3 fonctions identiques (aux entrées et sorties près) *ChangeDC1*, *ChangeDC2* et *ChangeDC4* permettent de changer le rapport cyclique des sorties associées respectivement à la LED verte, à la LED rouge et à la LED bleue. La valeur du rapport cyclique que peuvent prendre ces trois fonctions est compris en 0 et 255.

Par ailleurs, les trois fonctions *initPWM1*, *initPWM2* et *initPWM4* permettent d'initialiser les trois sorties PWM que nous utilisons.

Interruption par appui sur bouton poussoir :

Le changement de mode du dispositif est effectué à l'aide d'un appui sur un bouton poussoir. L'appui sur ce bouton poussoir déclenche une interruption sur changement d'état. Le programme d'interruption est codé dans la fonction *prog_interruption*. Notre dispositif pouvant fonctionner sous trois modes différents, nous avons pris soin dans la fonction d'interruption de donner une valeur comprise entre 0 et 2 à un compteur en fonction du mode sélectionné.

Corps du code :

Le coeur de notre algorithme se trouve dans la fonction main du code. C'est dans cette partie que sont implémentés les différents modes de notre dispositif.

Le cas par défaut permet d'implémenter le mode adaptatif : on récupère un signal numérique grâce au CAN (convertisseur analogique-numérique) issus des trois photodiodes, et on prend le complémentaire du signal qui code la valeur de l'intensité lumineuse ambiante pour chaque couleur (ie 255 - valeur du signal) comme valeur de modulation en largeur d'impulsions. Ainsi, si la luminosité ambiante est fortement coloré, le logo projeté sera coloré dans les couleurs complémentaires ce qui permet d'obtenir une meilleure visibilité et un meilleur contraste du logo.

Le cas 1 correspond quant à lui à la roue de couleur : on parcourt de manière régulière différentes couleurs du spectre. Pour cela on fait varier de manière périodique les différentes valeurs de modulation en largeur d'impulsion. Pour sélectionner une couleur il suffit d'appuyer sur le bouton poussoir au moment où la couleur passe dans la roue de couleur. Nous entrons alors dans le cas 2, et l'algorithme ne fait plus rien jusqu'à la prochaine interruption qui le refait passer en mode automatique.

La sortie vers la source

En sortie du PIC on a un signal PWM qui nous permet de contrôler l'intensité de chaque LEDs. Pour ce faire on utilise des transistors (IRF540N). La LED est branchée entre V_{cc} et la résistance (par exemple R5 pour la bleue). Lorsque V_s est à 5V, la LED est alimentée et lorsque V_s est à 0V, la LED n'est pas alimentée : le transistor se comporte comme une sorte d'interrupteur qui s'ouvre et se ferme en fonction de la tension envoyée sur sa broche G. Ainsi en branchant le rapport cyclique à la broche G on peut contrôler l'intensité transmise à la LED. En effet la LED ne va pas voir l'interrupteur qui s'ouvre et qui se ferme mais seulement une valeur moyenne d'intensité. Ainsi on contrôle précisément à l'aide d'un simple PWM et d'un transistor la puissance de la LED en sortie.

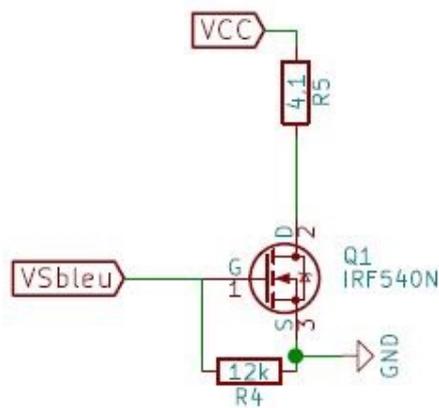


FIGURE 6 - Montage de modulation de la tension qui sera envoyée à la LED bleue, selon le signal PWM en entrée.

Bloc d'émission

La source lumineuse

Les premiers essais ont été réalisés avec une LED Luxeon K2 qui fonctionnait avec un courant de 1,5 A et pouvait atteindre jusqu'à 175 lumen. Cette LED n'éclairait qu'en lumière blanche mais nous a permis de réaliser qu'une LED de puissance était nécessaire, fonctionnant avec un courant de l'ordre de l'ampère. De plus, nous avons réalisé que le genre de LED que nous recherchions chauffait beaucoup d'où la nécessité de la monter sur un radiateur.

Le but de notre projecteur étant de s'adapter à son environnement en changeant la couleur projetée, il nous fallait évidemment des LED de couleurs différentes. Nous voulions également suffisamment d'intensité lumineuse pour former une image sur un écran à quelques mètres du projecteur. La solution que nous avons trouvée est la LED LZP-00MD00.

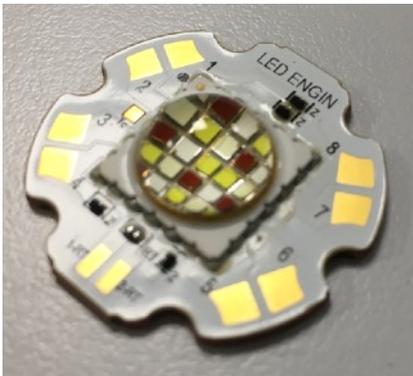


FIGURE 7 - LED LZP-00MD00.

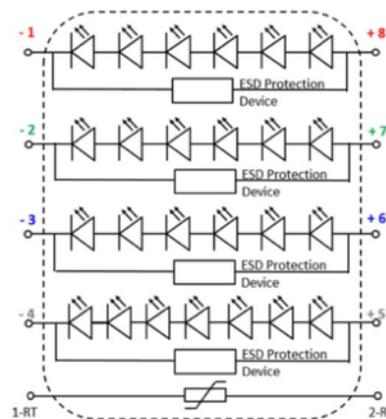


FIGURE 8 - Schéma de la LED
(issu de la documentation
technique)

Cette LED a une puissance de dissipation de 80W. Comme représentée à la figure 8, c'est en réalité une matrice de LEDs : elle contient en réalité 6 ou 7 LED en séries pour chaque couleur, afin d'assurer un important flux lumineux. Les données du constructeur pour une intensité $I_F = 1A$ indiquent :

- pour le rouge : $\Phi_V = 1060 \text{ lm}$
- pour le vert : $\Phi_V = 1190 \text{ lm}$
- pour le bleu : $\Phi_V = 300 \text{ lm}$
- pour le blanc : $\Phi_V = 2000 \text{ lm}$.

La LZP-00MD00 dispose d'un circuit dédié pour chaque canal de couleur, ce qui nous sera très utile pour contrôler la couleur de la projection. A cela s'ajoute la répartition des LED de couleur sur la matrice : cette position permet d'uniformiser la couleur globale de la LED. Le dôme surplombant la matrice perfectionne cette uniformisation, de telle sorte que nous n'avons même pas besoin d'ajouter un diffusant (comme une opaline par exemple) pour obtenir une seule image lors de la projection (et non pas une image par LED de la matrice).

La LED présente encore un avantage : elle ne chauffe pas trop pour une LED de puissance. Nous avons toutefois décidé de la monter sur un radiateur, pour faciliter la dissipation thermique.

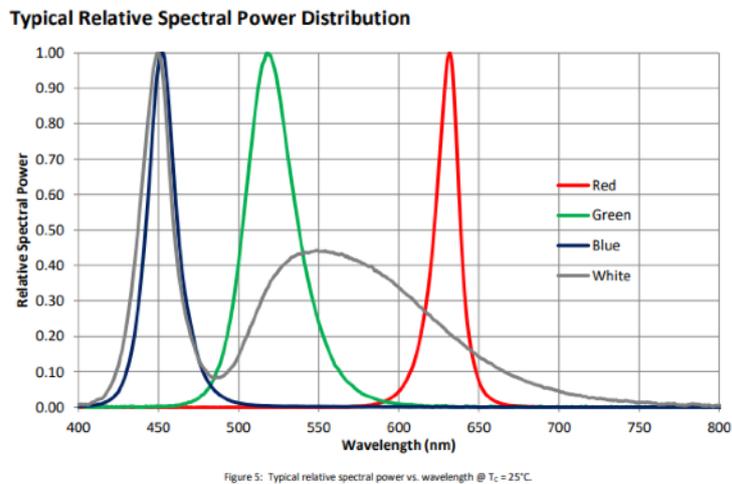


FIGURE 9 - Caractéristique spectrale de la LED LZP-00MD00 (issue de la documentation technique).

Pour connecter cette LED au circuit précédemment détaillé, il reste donc à connecter chaque canal de couleur à la sortie du transistor correspondant. Pour cela, il faut faire attention au fait que chaque canal fonctionne avec une tension différente : nous avons donc employé des résistances de protection pour abaisser la tension arrivant sur chacune des LED de couleurs.

Nous alimentons les LEDs en 24V. Pour chaque canal de couleur, il faut un courant de 1A et on peut lire sur la fiche technique $V_{\text{rouge}} = 16.2 \text{ V}$; $V_{\text{vert}} = 22.4 \text{ V}$; $V_{\text{bleu}} = 19.9 \text{ V}$ et $V_{\text{blanc}} = 23.3 \text{ V}$.

Avec la loi d'Ohm, on trouve les valeurs de résistance suivantes : $R_{\text{rouge}} = 7.8 \Omega$, $R_{\text{vert}} = 1.6 \Omega$, $R_{\text{bleu}} = 4.1 \Omega$ et $R_{\text{blanc}} = 0.7 \Omega$. Cependant, comme ces résistances seront traversées par une forte intensité ($4 \times 1\text{A} = 4\text{A}$), il faut utiliser des résistances de puissance.

Les éléments du bloc optique

Maintenant que la source est choisie et reliée au circuit, intéressons-nous à la partie optique du projecteur. Notre système optique se base sur un ancien appareil à diapositives, dont nous avons repris les différents éléments (il est représenté à la figure 10).

Les deux premières lentilles sont celles issues de l'appareil à diapositives du LEnsE. La dernière lentille (celle de l'objectif, à droite) a été déterminée pour l'utilisation de projection à des distances allant de 1m à quelques mètres. La distance diapositive-lentille est ajustable (nous avons repris le dispositif de l'ancien appareil) afin de régler la focale du projecteur.

La distance entre la LED et la première lentille a également été déterminée de manière expérimentale et correspond en fait à la distance qui permet d'avoir un éclairage Köhler. Dans le cas de l'illumination de Köhler, l'image de la source n'est pas superposée à celle de l'échantillon. Ainsi, chaque point de la source illumine tout l'objet, rendant l'illumination uniforme. Le réglage consiste à assurer la conjugaison des différents plans images de l'objet (diapositive) et la conjugaison des différents plans images de la source (source).

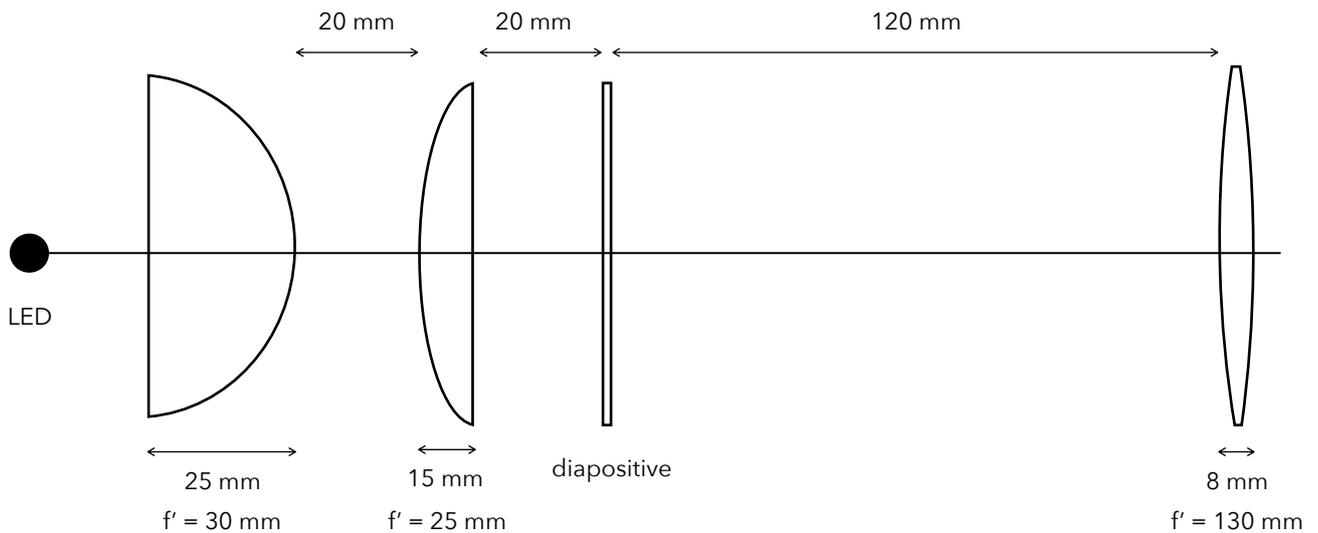


FIGURE 10 - Schéma du système optique du projecteur.

Boîtier d'alimentation

Pour pouvoir avoir un produit fini facile à déplacer et à alimenter, nous avons entrepris de concevoir un boîtier qui contiendrait la partie électronique et l'alimentation.

Nous avons ainsi réuni toutes la partie électronique (détection, traitement) sur un circuit imprimé.

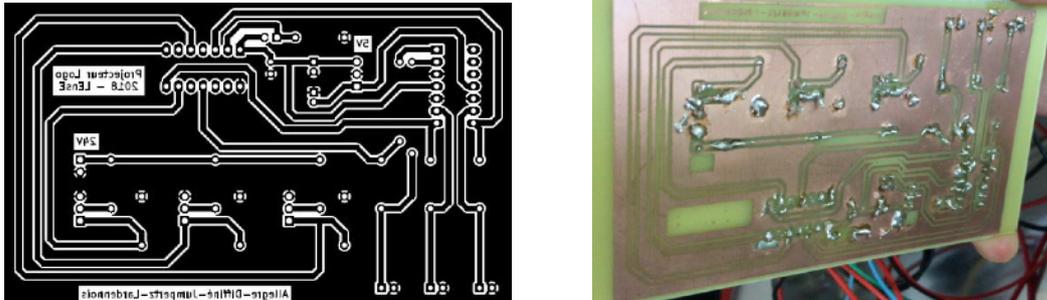


FIGURE 11 - Circuit imprimé regroupant les parties de détections, de traitement, et de sortie vers les LEDs. A gauche le schéma, à droite le résultat.

Nous nous étions fixé dans le cahier des charges, qu'il fallait pouvoir brancher le projecteur à une prise du secteur. Cette dernière envoie un courant alternatif mais notre circuit fonctionne en courant continu. Nous avons donc dû intercaler entre le secteur et les deux entrées en tension de notre système un convertisseur alternatif-continu. Nous avons choisi de sélectionner le composant FKH2- 50-04, de par son relativement faible coût.

Après ce convertisseur alternatif-continu, nous avons placé des alimentations 24V (alimentation 621 0685) et 15V (alimentation 621 0685) pour alimenter respectivement les LED et le reste du circuit.

En réalité, le circuit a été conçu pour des entrées à ± 5 V, mais un problème de communication entre les équipes nous a fait choisir une alimentation ± 15 V (nous envisagions au début d'alimenter l'ALI en ± 15 V et d'utiliser un régulateur de puissance pour alimenter le PIC en +5V, mais le circuit imprimé étant déjà effectué, nous avons été obligé d'alimenter l'ALI et le PIC avec la même tension). Nous avons donc finalement alimenté les deux à 5V en rajoutant deux régulateurs de puissance (LM7805 CV et LM 7905CT) déportés du circuit imprimé pour pouvoir transformer les ± 15 V de l'alimentation 466 6878 en ± 5 V.

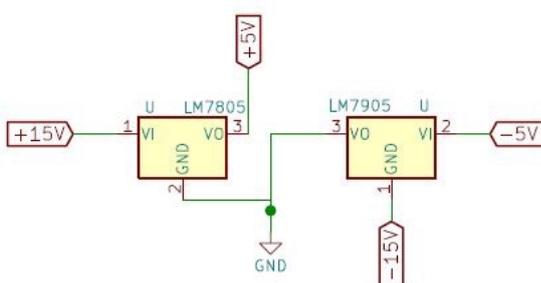
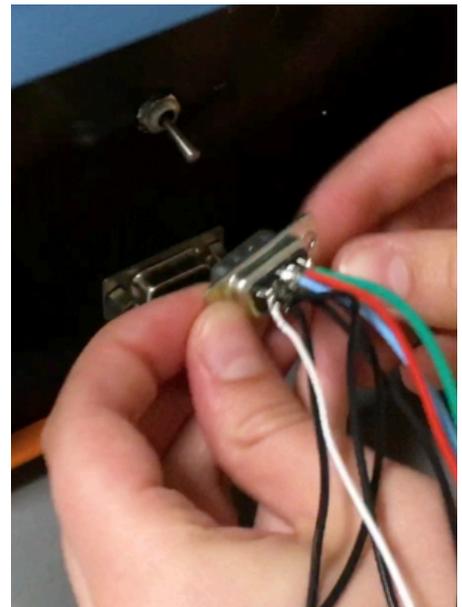
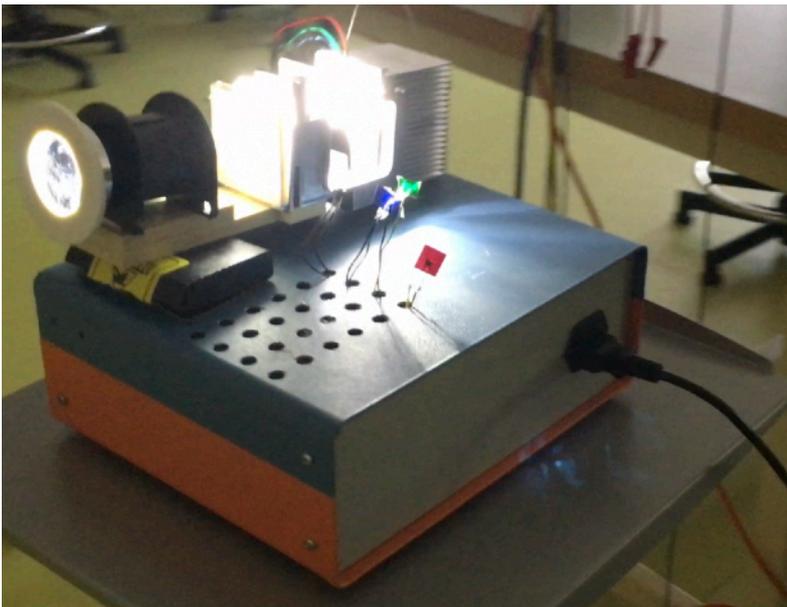


FIGURE 11 - Schéma de la régulation en tension (déportée du circuit imprimé). Le ± 15 V de l'alimentation est convertie en ± 5 V, qui servira à alimenter le circuit.

Conclusion

Le produit final que nous vous présentons est fonctionnel et nous avons pu constater son efficacité en le testant. Nous avons réalisé un projecteur de logo, portatif grâce à son boîtier, et qui s'adapte à la luminosité de son environnement. L'utilisateur de ce projecteur pourra ainsi bénéficier soit d'une lumière blanche classique, soit le choix parmi les trois modes qu'il décide grâce au bouton poussoir.

Nous avons conçu un circuit imprimé afin de transporter notre projecteur. Malheureusement, un problème de soudure en a empêché le bon fonctionnement. Ainsi, il faudrait revoir la réalisation de circuit afin qu'il puisse assurer le fonctionnement de notre projecteur. Aussi, il faudrait trouver une autre solution que les résistances de puissance devant les LEDS afin de ne pas perdre trop d'énergie.



Notice d'utilisation

Le fonctionnement de notre projecteur de logo qui s'adapte à son environnement est très simple.

Il suffit de brancher le boîtier au secteur puis allumer le projecteur en abaissant l'interrupteur ON/OFF (au dessus du connecteur de la LED).

Le projecteur est alors allumé et peut fonctionner. On peut changer le mode de fonctionnement en appuyant sur le bouton poussoir. Il existe trois modes différents :

- Le mode « adaptatif » : c'est le principal mode de notre projecteur. En fonction de la luminosité ambiante, le projecteur s'adapte pour renvoyer une image bien visible. Ainsi, le projecteur aura tendance à illuminer la diapositive avec une couleur complémentaire à la luminosité ambiante dans un souci d'amélioration du contraste.
- Le mode « roue chromatique » : le projecteur éclaire la diapositive avec des couleurs qui varient régulièrement. On peut sélectionner une de ces couleurs à l'aide d'un appui sur le bouton poussoir et alors passer dans le dernier mode.
- Le mode « manuel » : le projecteur éclaire la diapositive de la couleur que nous avons choisi dans le mode roue.

Le deuxième interrupteur sur le boîtier correspond à la LED blanche. On peut l'allumer indépendamment des LEDs de couleur. Cette fonctionnalité reprend celle d'un appareil à diapositives basique. Allumer la LEDs blanche permet d'éclairer la diapositive avec encore plus de puissance, mais risque de ne plus rendre visible l'adaptation en couleur de la diapositive.

Si on veut changer l'image projetée, on peut alors retirer la diapositive par le haut et en insérer une autre.

Pour régler la netteté du projecteur, il faut toucher à l'objectif. L'objectif a été monté sur un support adaptatif. On fait alors tourner l'objectif, ce qui modifie la distance entre l'objectif et la lentille, modifiant ainsi la conjugaison et permet alors d'adapter la netteté de l'image en fonction de la distance entre le projecteur et la surface que laquelle on veut projeter notre image.

Annexe : programme du PIC16F1503

```
#include <xc.h>
#include "../Ressources/config.h"
#include <stdio.h>
#include <stdlib.h>

void initPWM1(void);
void initPWM2(void);
void initPWM4(void);
void initADC(void);
void initPIC(void);
void changeDC1(char val);
void changeDC2(char val);
void changeDC4(char val);
void interrupt prog_interruption (void);

int compteur = 0;
int compteur_roue = 0;
int rouge, bleu, vert;

int main(){
    initPIC();
    initPWM1();
    initPWM2();
    initPWM4();
    initADC();
    compteur = 0;
    while (1){
        switch (compteur){
            case 1 :
                if (compteur_roue == 0){
                    rouge = 250;
                    bleu = 0;
                    vert = 0;
                    changeDC1(vert);
                    changeDC4(bleu);
                    changeDC2(rouge);
                }
                if (compteur_roue < 5){
                    vert = vert + 50;
                    changeDC1(vert);
                    compteur_roue = compteur_roue + 1;
                }
                if (compteur_roue < 10 && compteur_roue >= 5){
                    rouge = rouge - 50;
                    changeDC2(rouge);
                    compteur_roue = compteur_roue + 1;
                }
            }
        }
    }
}
```

```

if (compteur_roue < 15 && compteur_roue >= 10){
    bleu = bleu + 50 ;
    changeDC4(bleu) ;
    compteur_roue = compteur_roue + 1 ;
}
if (compteur_roue < 20 && compteur_roue >= 15){
    vert = vert - 50 ;
    changeDC1(vert) ;
    compteur_roue = compteur_roue + 1 ;
}
if (compteur_roue < 25 && compteur_roue >= 20){
    rouge = rouge + 50 ;
    changeDC2(rouge) ;
    compteur_roue = compteur_roue + 1 ;
}
if (compteur_roue < 30 && compteur_roue >= 25){
    bleu = bleu - 50 ;
    changeDC4(bleu) ;
    compteur_roue = compteur_roue + 1 ;
}
if (compteur_roue == 30){
    compteur_roue = 0 ;
}
__delay_ms(500);
break ;
case 2 :
    break ;
default :
    // ROUGE : Conversion analog-num photodiode rouge
    ADCON0bits.CHS = 3; //AN3 : RA4
    ADCON0bits.GO = 1;
    while(ADCON0bits.GO == 0);
    rouge = ADRES;
    // BLEU : Conversion analog-num photodiode bleu
    ADCON0bits.CHS = 2; //AN2 : RA2
    ADCON0bits.GO = 1;
    while(ADCON0bits.GO == 0);
    bleu = ADRES;
    // VERT : Conversion analog-num photodiode verte
    ADCON0bits.CHS = 6; //AN6 : RC2
    ADCON0bits.GO = 1;
    while(ADCON0bits.GO == 0);
    vert = ADRES;
    changeDC1(255 - vert) ;
    changeDC2(255 - rouge) ;
    changeDC4(255 - bleu) ;
}
}
}

```

```

void changeDC1(char val){ // VERT
    int temp = (val * 4 * (PR2+1)/255);
    PWM1DCL = temp << 6;
    PWM1DCH = temp >> 2;
}

void changeDC2(char val){ // ROUGE
    int temp = (val * 4 * (PR2+1)/255);
    PWM2DCL = temp << 6;
    PWM2DCH = temp >> 2;
}

void changeDC4(char val){ // BLEU
    int temp = (val * 4 * (PR2+1)/255);
    PWM4DCL = temp << 6;
    PWM4DCH = temp >> 2;
}

void initPWM1(void){ //VERT
    TRISCbits.TRISC5 = 0;
    PWM1CONbits.PWM1EN = 1; // active le module PWM
    PWM1CONbits.PWM1OE = 1; // active la sortie du PWM
    PWM1CONbits.PWM1POL = 0; // PWM output is active high
    PR2= 31;
    T2CONbits.T2CKPS = 0b10;
    T2CONbits.TMR2ON = 1;
    changeDC1(0);
}

void initPWM2(void){ //ROUGE
    TRISCbits.TRISC3 = 0;
    PWM2CONbits.PWM2EN = 1; // active le module PWM
    PWM2CONbits.PWM2OE = 1; // active la sortie du PWM
    PWM2CONbits.PWM2POL = 0; // PWM output is active high
    PR2 = 31;
    T2CONbits.T2CKPS = 0b10;
    T2CONbits.TMR2ON = 1;
    changeDC2(0);
}

void initPWM4(void){ //BLEU
    TRISCbits.TRISC1 = 0;
    PWM4CONbits.PWM4EN = 1; // active le module PWM
    PWM4CONbits.PWM4OE = 1; // active la sortie du PWM
    PWM4CONbits.PWM4POL = 0; // PWM output is active high
    PR2 = 31;
    T2CONbits.T2CKPS = 0b10;
    T2CONbits.TMR2ON = 1;
    changeDC4(0);
}

```

```

void initADC(void){
    TRISAbits.TRISA2 = 1;
    ANSELAbits.ANSA2 = 1;
    TRISAbits.TRISA4 = 1;
    ANSELAbits.ANSA4 = 1;
    TRISCbits.TRISC2 = 1;
    ANSELCbits.ANSC2 = 1;
    ADCON0bits.CHS = 5;
    ADCON1bits.ADFM = 1;
    ADCON1bits.ADPREF = 0b00;
    ADCON1bits.ADCS = 0b011;
    ADCON0bits.ADON = 1;
}

```

```

void initPIC(void){
    ANSELA = 0;
    ANSELC = 0;
    OSCCONbits.IRCF = 0b1100;
    TRISAbits.TRISA5 = 1 ;
    IOCAP = 0;
    IOCAN = 0;
    IOCAPbits.IOCAP5 = 1;
    INTCON = 0;
    INTCONbits.IOCIE = 1;
    INTCONbits.GIE = 1;
    return;
}

```

```

void interrupt prog_interruption (void){
    if (INTCONbits.IOCIF == 1){
        if (IOCAFbits.IOCAF5 == 1){
            compteur = (compteur + 1) % 3;
            __delay_ms(100);
            IOCAFbits.IOCAF5 = 0 ;
        }
    }
}

```