

Projet d'Électronique

Jeu de mémoire visuel et sonore « Simon »

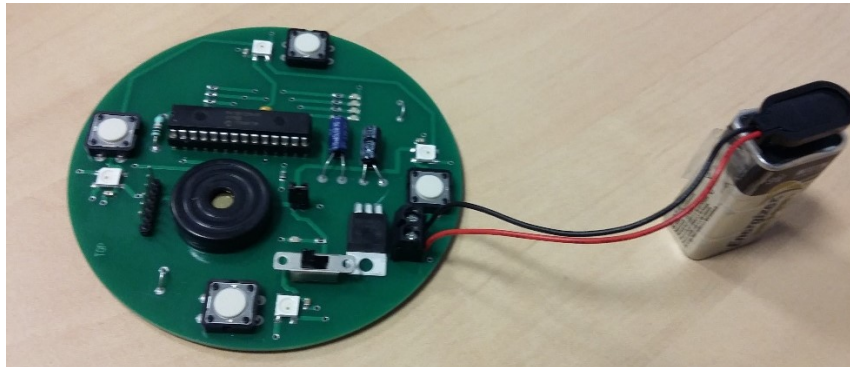
1. Objectifs du projet

L'objectif de ce projet est double :

- Il permet de mettre au point, assembler et tester un montage électronique dont la fonction peut aisément être mise en évidence. Dans la continuité de la formation théorique, le but recherché est de mettre en pratique les connaissances acquises en électronique numérique, et faire découvrir par l'expérimentation les principales étapes de développement d'un système électronique.
- Il est représentatif des systèmes numériques actuels, pour lesquels on trouve :
 - l'acquisition de signaux issus de l'environnement,
 - le traitement numérique de l'information,
 - la mise en œuvre d'actionneurs en fonction de l'information reçue.

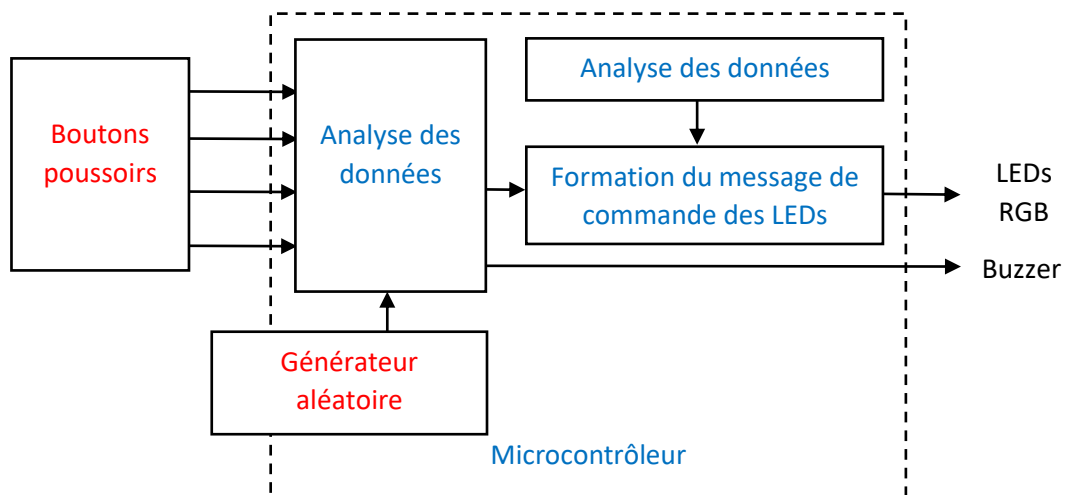
2. Présentation du sujet

Le projet consiste à concevoir le circuit dérivé du jeu « Simon ». Ce jeu consiste à générer aléatoirement une séquence de sons et de couleurs à l'aide d'un buzzer et de 4 LEDs RGB (Red-Green-Blue) adressables individuellement. A l'aide de boutons poussoir, il s'agit ensuite de reconstituer la séquence dans l'ordre. Si on y arrive, la séquence se complexifie au fur et à mesure, et si on n'y arrive pas, la séquence est remise à zéro.



Le jeu pourra être alimenté par un générateur de tension, mais il est également prévu de pouvoir l'alimenter par une pile 9V.

Le schéma-bloc suivant montre le fonctionnement de principe de l'ensemble du système :



3. Cahier des charges

Le circuit complet est alimenté par une tension V_{DD} de 5V. Certains composants sont fragiles, donc cette valeur de tension devra être scrupuleusement respectée.

Le système s'articule autour d'un microcontrôleur MICROCHIP de référence PIC18F25K40 dans le circuit.

Le programme du microcontrôleur doit être développé en assembleur.

ATTENTION : beaucoup de registres nécessaires à la configuration des fonctionnalités utilisées dans le projet sont situés hors de l'Access Bank. Il faudra donc faire attention lors de la configuration des différents modules à la localisation dans la mémoire des registres utilisés. Vous pourrez trouver les informations sur la position des différents registres dans la mémoire aux pages 101 et 102 de la datasheet.

Les LEDs utilisées ont pour référence SK6812RGBW. Ces LEDs utilisent un protocole particulier pour recevoir de l'information en série, peu importe leur nombre. Ce protocole devra être imité en utilisant **une seule sortie** du microcontrôleur. Plus d'informations sur ce protocole sont données en annexe de ce document.

ATTENTION : ces LEDs sont extrêmement fragiles ! Elles sont facilement mises hors d'usage si elles sont mal connectées ou soudées, merci de faire attention à leur branchement. Plus de détails sont donnés en annexe.

Le buzzer est contrôlé par un signal PWM provenant du microcontrôleur. La fréquence de ce signal détermine la hauteur (« pitch ») du son émis.

Un générateur de nombre aléatoire (ou au moins pseudo-aléatoire) devra être mis en place dans le programme du microcontrôleur. À vous de déterminer le moyen le plus adapté pour le concevoir, parmi les possibilités natives qu'offre le microcontrôleur.

Quatre pattes du microcontrôleur sont prévues pour pouvoir commander de simples LEDs vertes individuelles supplémentaires montées directement sur la carte électronique. Elles peuvent être utilisées selon le bon vouloir des concepteurs :

- pour indiquer que le système est sous tension,
- pour afficher un timer de réponse,
- pour afficher un rappel de la position des LEDs allumées,
- etc.

Toute autre proposition intéressante est bienvenue !

4. Matériel fourni

4.1. Appareils de mesure fournis

Chaque groupe de projet travaillera sur une paillasse de test. Sur chaque paillasse, plusieurs appareils de mesure sont mis à disposition pour les différents tests à effectuer :

- 1 oscilloscope 2 voies minimum
- 1 générateur DC double voie (ou 2 générateurs DC simple voie)
- 1 générateur de signaux basse fréquence (GBF)

De plus, pour connecter les montages aux appareils de mesure, chaque étudiant aura à sa disposition un kit personnel contenant :

- 1 breadboard (platine de connexion entre composants)
- 1 sonde d'oscilloscope
- 2 câbles « banane » courts (1 rouge et 1 noir)
- 2 câbles « banane » longs (1 rouge et 1 noir)
- 4 fiches « banane » démontables
- 1 tournevis

Il est à noter que ce kit sera distribué uniquement aux étudiants de CSI3 non doublants. Les autres étudiants devront apporter le kit qui leur a déjà été donné dans leur scolarité.

D'autres types de câble (BNC vers « banane » ou câbles jack par exemple) peuvent être utilisés sur demande à un encadrant de projet.

Pour vérifier et mesurer des valeurs de résistances ou de capacités, des multimètres et un impédancemètre sont à disposition, sur demande à un encadrant de projet.

4.2. Composants fournis en début de projet

En début de projet, les composants clairement indiqués sur les schémas vous sont fournis. Ils sont récapitulés dans la liste suivante.

1 circuit imprimé à souder

1 microcontrôleur MICROCHIP PIC18F25K40 + son support DIL 28 broches

4 boutons poussoirs

4 LEDs RGBW SK6812RGBW

(Composant Monté en Surface, ou Surface Mounted Device)

1 support droit mâle 6 contacts

1 buzzer

4 condensateurs 100nF – package 0805

(Composant Monté en Surface, ou Surface Mounted Device)

5 LED vertes – package 0805

(Composant Monté en Surface, ou Surface Mounted Device)

10 résistances 2.2k Ω – package 0805

(Composant Monté en Surface, ou Surface Mounted Device)

1 condensateur 100nF non polarisé (technologie : céramique)

1 résistance 10k Ω

+ résistances et condensateurs divers...

4.3. Composants fournis en fin de projet (optionnel)

Les composants suivants pourront être ajoutés après validation des encadrants pour pouvoir brancher le circuit sur une pile 9V :

1 régulateur de tension 5V 1A de référence L7805ABV

1 commutateur glissant rectiligne SPDT 2 positions

1 bornier 2 contacts

1 condensateur 10 μ F / 63V polarisé (technologie : électrolyte)

1 condensateur 1 μ F / 63V polarisé (technologie : électrolyte)

The diagram illustrates the hardware setup for a PIC18F73K40 microcontroller. It includes power supply regulation, pin configurations for various peripherals, and signal processing components.

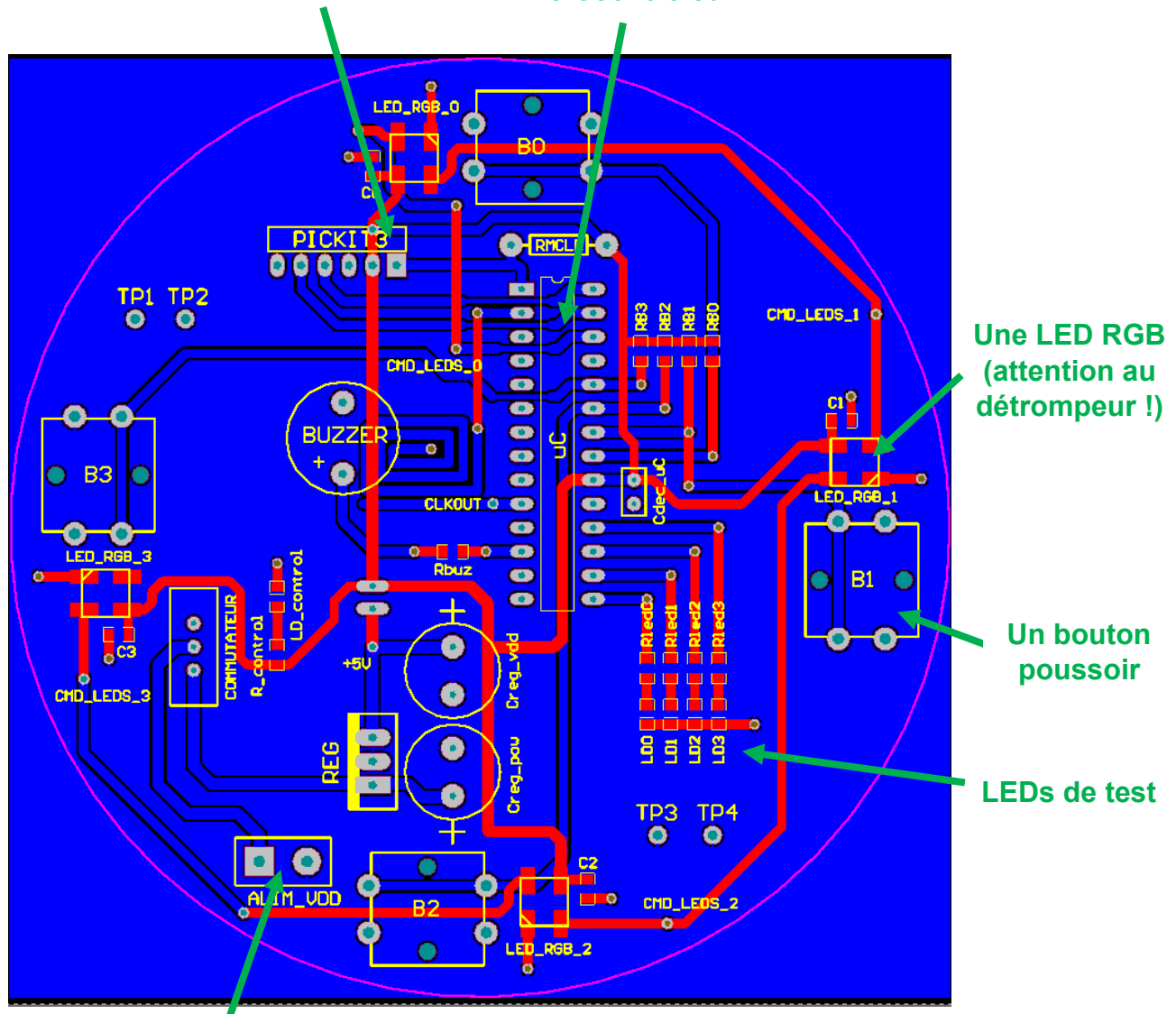
- Section A:** Power supply section showing V_{DD}, GND, and various decoupling capacitors (C_{dec}, C_{reg}) connected to the microcontroller's power pins.
- Section B:** Microcontroller pin configuration. The PIC18F73K40 is shown with its pins connected to various components:
 - Pins RB0-RB7 are connected to LEDs (LED0-LED7) through current-limiting resistors (R_{led0}-R_{led7}).
 - Pins RA0-RA7 are connected to a buzzer (BUZZER) through a resistor (R_{buz}).
 - Pins RC0-RC7 are connected to a potentiometer (POT) through resistors (R_{p0}-R_{p7}).
 - Pins RD0-RD7 are connected to a potentiometer (POT) through resistors (R_{d0}-R_{d7}).
- Section C:** LED driver circuit. This section shows the connection of LEDs (LED0-LED7) to the microcontroller's output pins (RB0-RB7) through current-limiting resistors (R_{led0}-R_{led7}). It also includes a common ground connection for all LEDs.
- Section D:** Buzzer and Potentiometer connections. This section shows the connection of a buzzer (BUZZER) and a potentiometer (POT) to the microcontroller's input/output pins (RA0-RA7 and RC0-RC7) through resistors (R_{buz} and R_{p0}-R_{p7}).

6. Disposition des composants sur les circuits imprimés

L'implantation des composants sur le circuit imprimé est la suivante (également disponible sur le groupe Teams du projet) :

Branchement du programmeur (« flèche » du PicKit3 sur la droite)

Microcontrôleur



Alimentation autonome de la carte

Remarque : Les points de test TP1 à TP4 servent à souder des points de connexion pour à se connecter à la masse du circuit, par exemple avec la masse de l'oscilloscope.

7. Méthodologie – Validation

Le travail est à effectuer en binôme.

ATTENTION : en fin de projet, vous devrez être en mesure de comprendre et d'expliquer l'ensemble du travail de votre groupe de projet, pas seulement le vôtre !

La structure du montage n'est pas à concevoir : elle vous est donnée, mais c'est à vous de comprendre le principe de fonctionnement et d'expliquer le rôle des différents éléments.

Les valeurs de certaines capacités sont fixées. **Attention au sens de montage des condensateurs polarisés !**

Le circuit sera monté sur un circuit imprimé double face (fourni en début de projet).

Vous devez mettre au point le montage et dimensionner les différentes parties du programme du microcontrôleur afin de remplir les fonctions souhaitées.

Remarques :

- **Il faut souder les composants CMS (montés en surface) avant tous les autres.** Ce sont en effet les plus petits composants, qu'il sera plus facile de souder sur un circuit imprimé vierge.
- **Il vaut mieux souder les composants permettant le bon fonctionnement du microcontrôleur en début de projet.** Vous aurez ainsi une plateforme de programmation plus fiable et moins sujette aux faux contacts que sur une plaquette d'essai. Ces composants sont :
 - Le support DIL28 pour pouvoir y enficher le microcontrôleur,
 - La résistance de 10kΩ branchée sur le Master Reset,
 - Le condensateur de découplage de 100nF,
 - Le support de programmation pour brancher le PicKit.
- **Il est déconseillé de tester le programme du microcontrôleur « d'un seul bloc ».** Mieux vaut développer des programmes simples de test qui seront faciles à valider avant de les compléter.
- **Dans un premier temps, l'alimentation du circuit sera à fournir par le programmeur. Une alimentation externe ne sera autorisée à être apportée qu'une fois que le reste du circuit sera fonctionnel.**

En fin de projet, une validation du circuit sera effectuée avec des encadrants : vous présenterez et expliquerez le dimensionnement de votre montage durant la phase de recette. À vous de montrer de la façon la plus convaincante que votre système fonctionne correctement et que vous avez compris le rôle de tous les éléments. Vous aurez accès pour cela à tout le matériel que vous avez pu utiliser durant la semaine de projet : générateurs, oscilloscope, etc.

Cette phase de validation aura lieu pendant la dernière demi-journée de projet pour tous les binômes. Par souci d'équité entre les groupes, le travail sur le projet s'arrêtera le vendredi matin.

8. Annexe 1 : édition des bits de configuration dans le PIC18F25K40

Afin de charger une configuration par défaut dans le PIC18F25K40 permettant son bon fonctionnement, il est nécessaire d'éditer les bits de configuration du microcontrôleur. Pour ce faire, vous pourrez vous reporter à la fin du sujet du TP Microcontrôleur de novembre/décembre (annexe 4). Les bits à éditer pour le PIC18F25K40 sont les suivants :

- CONFIG1L :

FEXTOSC = OFF ; Pas d'oscillateur externe

RSTOSC = n'importe quelle source d'oscillateur interne

- CONFIG3L :

WDTE = OFF

- CONFIG4H :

LVP = OFF

Les autres bits de configuration peuvent être laissés dans leur état par défaut.

Si vous souhaitez changer la fréquence d'horloge du microcontrôleur plus tard dans le programme, il suffit de changer le contenu du registre OSCFRQ (attention, il n'est pas dans l'Access Bank !). Sinon, si la valeur choisie par défaut vous convient, il n'y a rien à faire de plus.

9. Annexe 2 : exemple de configuration de l'ADC dans le PIC18F25K40

Le module ADC du PIC18F25K40 contient beaucoup plus d'options, plus ou moins intéressantes dans notre cas d'étude, que celui des microcontrôleurs PIC18F45K20 utilisé lors des séances de TP précédentes. Afin de rendre son utilisation plus aisée, vous trouverez ci-dessous un exemple de code permettant la configuration du module ADC, à compléter selon vos besoins (les bits à compléter sont indiqués par des `_`). Vous pourrez vous référer pour cela aux pages 441 à 457 de la datasheet du PIC18F25K40. Vous trouverez de plus un exemple de code de conversion à la page 430. Attention, ce code est générique pour toute la famille des PIC18. Si un commentaire de ce code est contradictoire avec la description d'un registre dans la datasheet, c'est cette dernière qu'il faut privilégier.

; début de la configuration

```
MOVLB    0x__      ; sélection de la banque d'adresse
MOVLW    b'_____'
MOVWF    ADPCH, 1    ; sélection du channel ADC
MOVLW    b'000_00__'
MOVWF    ADREF, 1    ; configuration des références analogiques
MOVLW    b'_____'
MOVWF    ADCLK, 1    ; configuration de l'horloge de l'ADC :  $1\mu s \leq T_{AD} \leq 6\mu s$ 
MOVLW    b'11111111'
MOVWF    ADPRE, 1    ; configuration du temps de précharge (max. par défaut)
MOVWF    ADACQ, 1    ; configuration du temps d'acquisition (max. par défaut)
CLRF     ADCAP, 1    ; pas de capacité additionnelle
MOVLW    b'00000000'
MOVWF    ADACT, 1    ; pas d'activation auto. de l'ADC sur événement
MOVLW    b'00000000'
MOVWF    ADCON3, 1   ; pas d'interruption sur l'ADC
MOVLW    b'00000000'
MOVWF    ADCON2, 1   ; configuration de l'ADC en mode basique
MOVLW    b'00000000'
MOVWF    ADCON1, 1   ; configuration des options de précharge
MOVLW    b'_0000_00'
MOVWF    ADCON0      ; configuration générale et format du résultat
```

; fin de la configuration

10. Annexe 3 : exemple de configuration d'un signal PWM dans le PIC18F25K40

Le module PWM du PIC18F25K40 contient beaucoup plus d'options, plus ou moins intéressantes dans notre cas d'étude, que celui des microcontrôleurs PIC18F45K20 utilisé lors des séances de TP précédentes. Afin de rendre son utilisation plus aisée, vous trouverez ci-dessous un exemple de code permettant la configuration du module CCP2 en mode PWM sortant sur la patte RC1, à compléter selon vos besoins (les bits/valeurs à compléter sont indiqués par des _). Vous pourrez vous référer pour cela aux pages 262 à 273 de la datasheet du PIC18F25K40. Vous trouverez de plus la succession des étapes à suivre à la page 271.

; début de la configuration

```
MOVLW    b'00000100'
MOVWF    CCPTMRS           ; associe le module CCP2 avec le timer 2
MOVLB    0x__             ; sélection de la banque d'adresse
MOVLW    0x06
MOVWF    RC1PPS, 1        ; associe le pin RC1 avec la fonction de sortie de CCP2
BSF      TRISC, 1         ; désactivation de la sortie PWM pour configuration
MOVLW    b'_____'
MOVWF    T2PR             ; fixe la période de PWM (voir formule p.271)
MOVLW    b'_00_____'
MOVWF    CCP2CON          ; configuration du module CCP2 et format des données
MOVLW    d'_____'
MOVWF    CCPR2H
MOVLW    d'_____'
MOVWF    CCPR2L           ; fixe le rapport cyclique du signal (voir formule p.272)
MOVLW    b'00000001'
MOVWF    T2CLKCON         ; configuration de l'horloge du timer 2 = Fosc/4
MOVLW    b'_____'
MOVWF    T2CON            ; choix des options du timer 2 (voir p.256)
BCF      TRISC, 1         ; activation de la sortie PWM
```

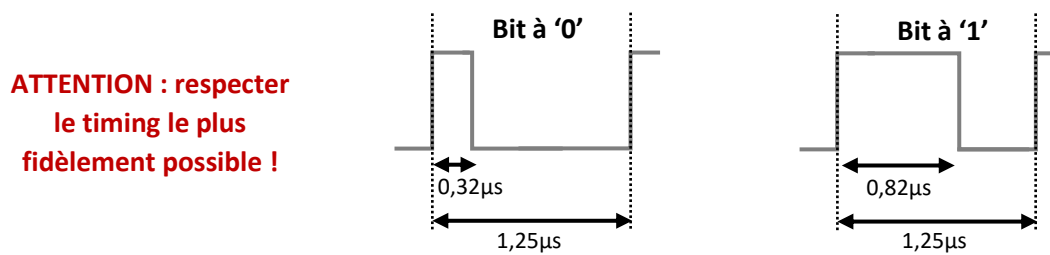
; fin de la configuration

11. Annexe 4 : adressage des LEDs RGBW

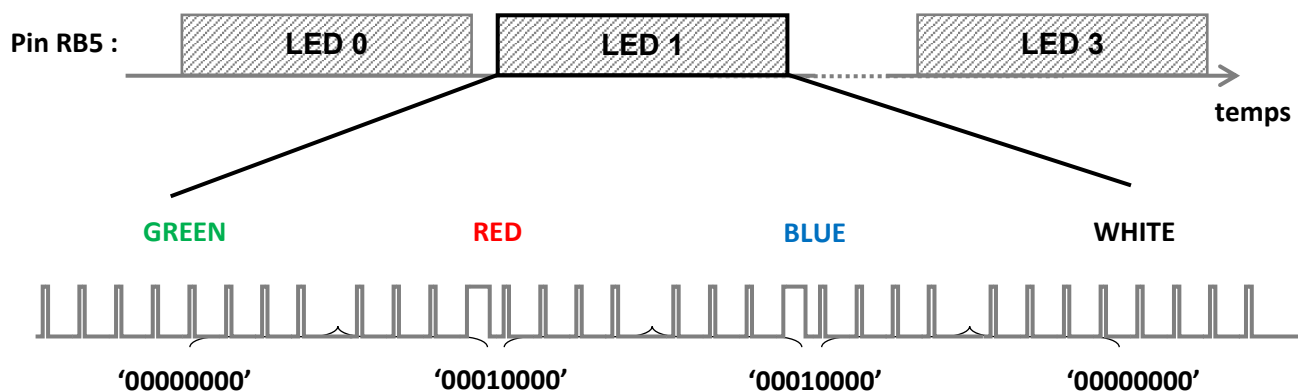
Les LEDs RGBW utilisées ont pour référence SK6812RGBW. Dans chaque boîtier de LED RGBW se trouvent 4 LEDs individuelles (une rouge, un vert, une bleue et une blanche), et un système gérant leur intensité. En mélangeant les couleurs de chacune des LEDs individuelles, il est possible d'obtenir une multitude de couleurs globales pour la LED RGBW. L'intensité des couleurs des LEDs individuelles est codée sur 8 bits, de la valeur 0 (LED éteinte) à la valeur 255 (LED complètement allumée). Il faut donc commander chaque LED RGBW sur 32 bits, ce qui donne 128 bits pour envoyer une commande aux 4 LEDs. Il n'y a besoin d'envoyer tous les bits de commande qu'une seule fois pour mettre à jour l'ensemble des LEDs, c'est-à-dire qu'il n'y a pas besoin de le répéter en boucle.

Ces 128 bits sont envoyés en série (c'est-à-dire via un seul signal, étalés dans le temps) sur la patte RB5 du microcontrôleur selon le protocole décrit dans le paragraphe suivant.

Chaque bit est codé par une impulsion, c'est-à-dire une succession d'un état haut puis d'un état bas du signal de commande pendant un certain temps. La somme des temps à l'état haut et à l'état bas reste identique pour coder un bit à '0' ou à '1', la différence se fait sur la répartition entre état haut et état bas, comme le montre le schéma suivant :



Le signal de commande de la matrice a donc l'allure suivante :



Dans l'exemple ci-dessus, la LED n°1 s'allume en violet (un peu de rouge + un peu de bleu).

Attention : La luminosité de ces LEDs est très forte (et la LED est donc gourmande en énergie). Dans le signal de commande, ne pas envoyer la valeur '255' pour allumer une couleur de la LED, mais préférer des valeurs de l'ordre de grandeur de quelques dizaines (les valeurs '16' ou '32' fonctionnent très bien).