

# Contrôle de la LED WS2812 à partir d'un DMA



Fabien Vannel  
Isn, hepia

# Protocole de la LED WS2812

La documentation complète de la LED RGB WS2812 fait référence.  
Les extraits suivants ont pour but de clarifier certains aspects du protocole.

Les 24 bits envoyés sur la LED codent les 3 intensités de lumière pour le vert, rouge et bleu. Ces bits sont envoyés successivement sur la ligne série dans l'ordre suivant:

## Composition of 24bit data:

G7	G6	G5	G4	G3	G2	G1	G0	R7	R6	R5	R4	R3	R2	R1	R0	B7	B6	B5	B4	B3	B2	B1	B0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

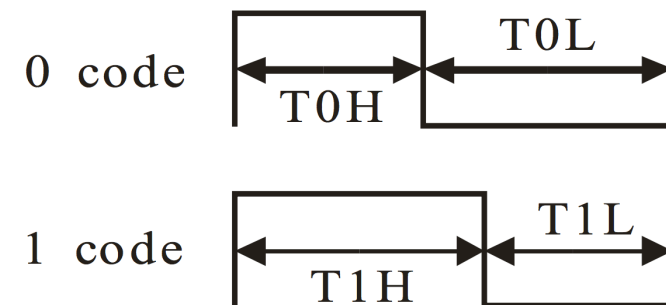
Note: Follow the order of GRB to sent data and the high bit sent at first.

# Codage d'un bit

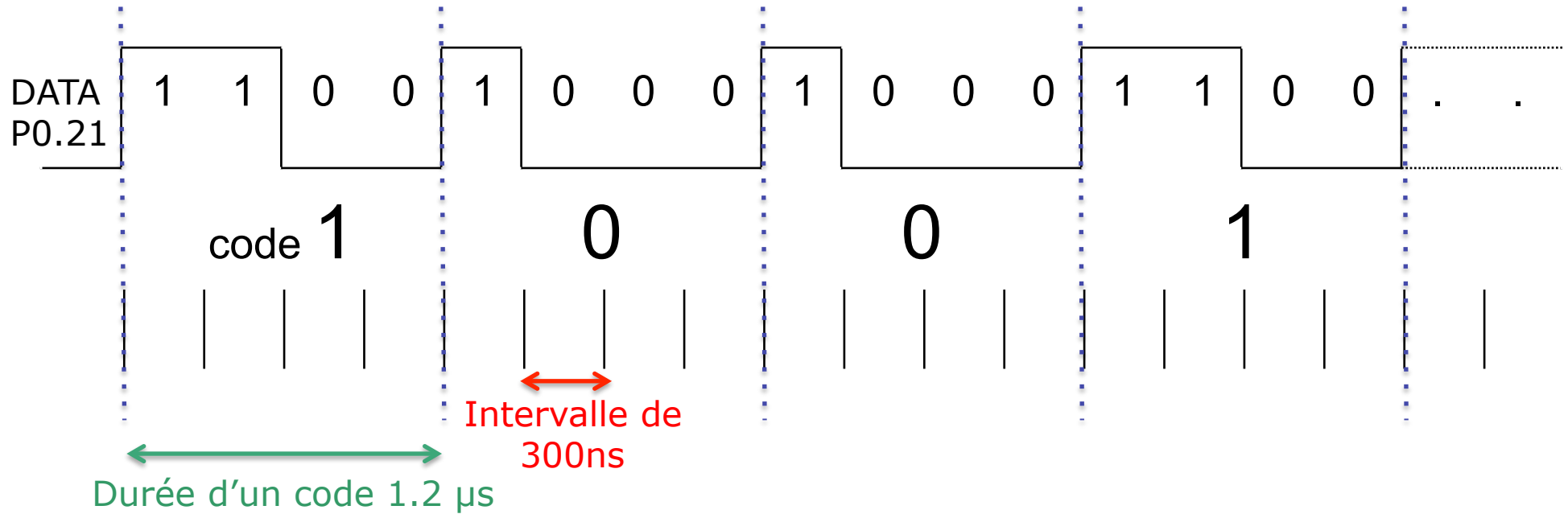
Chacun des 24 bits à transmettre est codé par un des codes ci-dessous.  
La durée de transmission d'un bit doit respecter une durée d'environ 1200 ns

**Data transfer time**(  $T_H + T_L = 1.25\mu s \pm 600ns$ )

T0H	0 code ,high voltage time	0.35us	$\pm 150ns$
T1H	1 code ,high voltage time	0.7us	$\pm 150ns$
T0L	0 code , low voltage time	0.8us	$\pm 150ns$
T1L	1 code ,low voltage time	0.6us	$\pm 150ns$
RES	low voltage time	Above 50 $\mu s$	



# Exemple de trame



- Une trame peut se diviser en subdivision de 300ns. Ainsi un code est transmis en  $4 \times 300\text{ns}$ , soit  $1.2\text{ }\mu\text{s}$ . Cette durée est conforme à la norme.
- Un code 1 est ainsi transmis en envoyant le signal DATA à l'état haut durant  $2 \times 300\text{ns}$ , puis  $2 \times 300\text{ ns}$  à l'état bas. (la spécification tolère  $700\text{ns} \pm 150\text{ns}$  pour l'état haut et  $600\text{ns} \pm 150\text{ns}$  pour l'état bas)
- Un code 0 est ainsi transmis en envoyant le signal DATA à l'état haut durant  $1 \times 300\text{ns}$ , puis  $3 \times 300\text{ ns}$  à l'état bas. (la spécification tolère  $350\text{ns} \pm 150\text{ns}$  pour l'état haut et  $800\text{ns} \pm 150\text{ns}$  pour l'état bas)

# Préparation des données

Tableau de 97 éléments:  
24 bits de couleurs x 4 données  
+ 1 bit (première donnée du  
tableau à 0)

Ce tableau ne pouvant pas être  
de type « bit », nous allons  
utiliser la représentation la plus  
petite, à savoir l'octet.

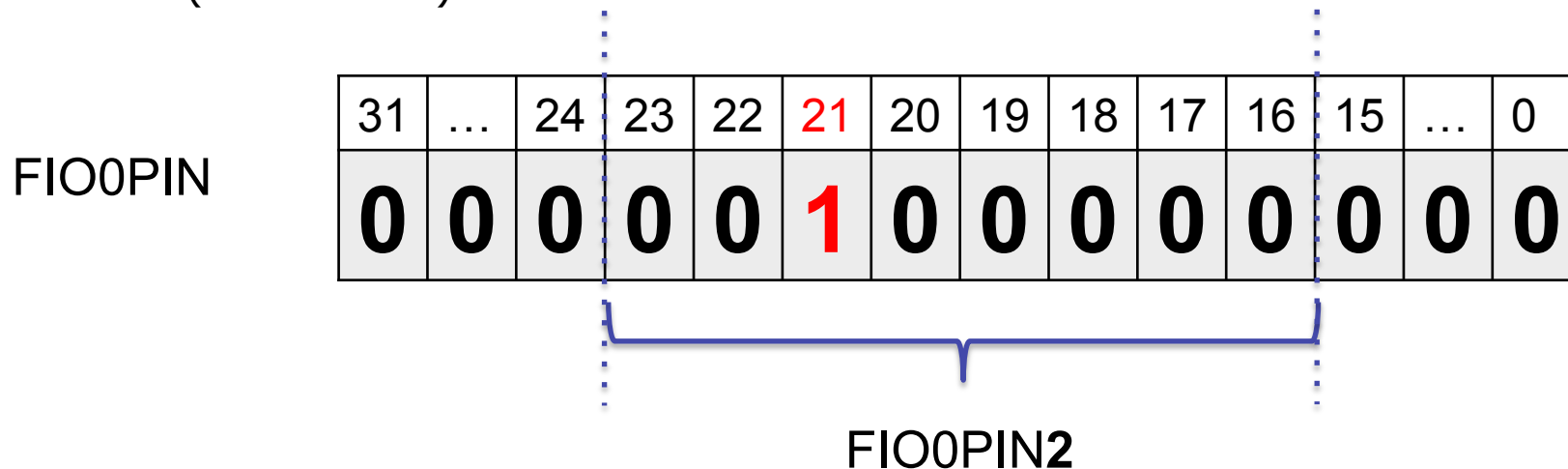
**uint8\_t tab[97];**

	tab [0]	0
	tab [1]	1
	tab [2]	1
	..	0
		0
		1
		0
		0
		0
		1
		0
		0
		0
		1
		1
		0
		0
	... tab [96]	...

# Contrôle de la LED

La LED RGB WS2812 est connectée sur la sortie du micro-contrôleur P0.21

Pour transmettre la bonne couleur à la LED, il suffit de périodiquement (toutes les 300 ns) envoyer la valeur contenue dans un élément du tableau sur le registre permettant de piloter la LED (ici FIO0PIN)



Le registre FIO0PIN2 permet de n'accéder qu'aux 8 bits allant des bits 23 à 16 du registre FIO0PIN. Ceci permet de ne donner que 8 bits de données au lieu des 32.

Ainsi dans le tableau précédent, il suffit de mettre correctement la valeur du bit de la LED souhaité au bit 5 dans les données du tableau