

Informatique embarquée

TD — GPIO

F.Ghaffari

S.Zuckerman

Objectifs

- Savoir programmer les GPIO et savoir les utiliser.

Matériel requis.

- Une plateforme nucleo-board STM32F446RE
- Une carte fille avec des LEDS

1 Interaction bouton-LED

On désire écrire un programme simple qui fait changer l'état de la LED de la carte (LED1) à chaque fois qu'on appuie sur le bouton USER_BUTTON (lui aussi sur la carte). L'annexe en fin d'énoncé contient une liste de fonctions qui peuvent être utiles à la réalisation du programme.

En utilisant les types `DigitalIn` et `DigitalOut`, écrire un programme qui va changer l'état de la LED de la carte (LED1) lorsqu'on appuie sur le bouton de la carte (USER_BUTTON). Ainsi, si la LED était allumée, appuyer sur USER_BUTTON fera qu'elle devra s'éteindre, et si elle était éteinte, le programme devra l'allumer.

1. Aller sur <https://www.mbed.com/> et créer un nouveau projet.
2. Écrire un programme qui allume la LED de la carte, LED1, pendant 1 seconde, avant de l'éteindre à nouveau. Le code qui allume et éteint la LED doit être contenu dans une fonction à part.
3. Modifier le programme précédent pour qu'il double la durée d'illumination de la LED à chaque nouvel appui sur le bouton. Écrire le nouveau code dans une fonction à part (différente de la fonction utilisée dans la question précédente).
4. Modifier le programme initial pour qu'il change l'état de la LED à chaque pression sur le bouton USER_BUTTON. La LED *reste* dans cet état jusqu'à la prochaine pression sur le bouton.

2 Programmation d'un chenillard

On souhaite réaliser un chenillard avec les GPIOs de type K2000.



FIGURE 1 – Kit, la voiture du futur.

1. Créer un nouveau projet dans MBED.
2. Connecter la carte fille fournie sur le connecteur CN10. Nous utiliserons par la suite les sorties D0 à D7 (voir Figure 2).
3. Programmer la carte pour allumer les leds D0 à D7 successivement. Lorsque D7 sera allumée, le sens d’affichage sera inversé (de D7 à D0). La durée d’affichage sera de 1s par LED.
4. On souhaite que la vitesse de défilement soit variable. Pour cela, un appui sur le bouton USER permettra de diviser par 2 la vitesse jusqu’à un ratio de 16. Arrivé à 16, le ratio sera remis à 1.

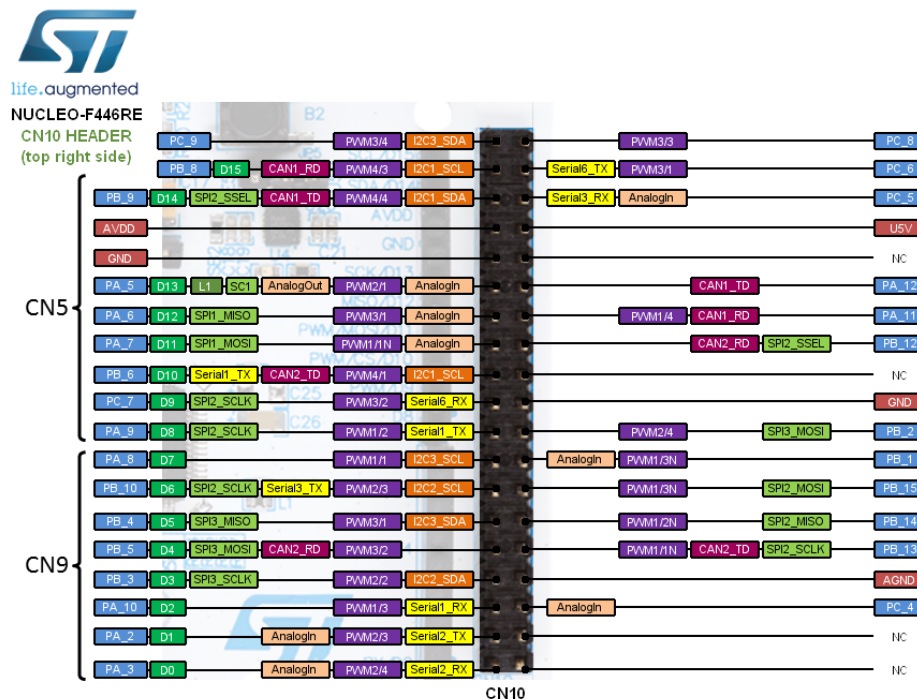


FIGURE 2 – Schéma des entrées-sorties pour la carte Nucleo-64.

Annexe — Liste de fonctions utiles

- `void wait(float delay);` force le micro-processeur à attendre `delay` micro-secondes avant d’exécuter l’instruction suivante dans le programme.