

UNIVERSITE de TOULOUSE

Faculté Sciences et Ingénierie

Master EEA

M1 SME

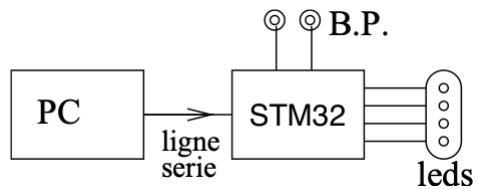
TP Micro-contrôleur

N. Rivière

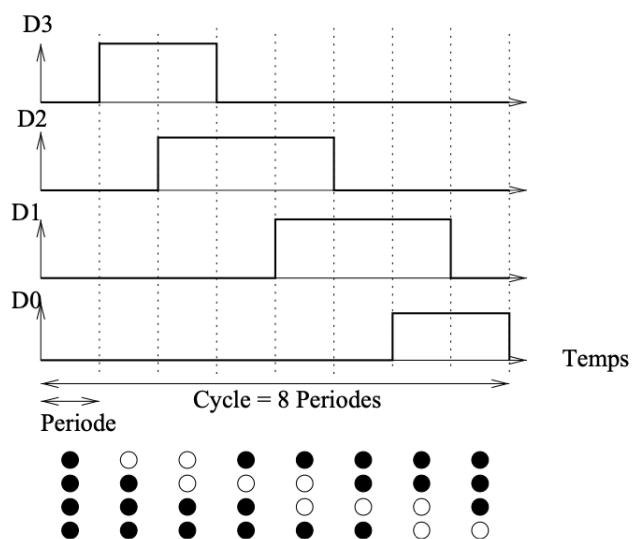
2025/2026

TP 1 : Séquenceur programmable

Introduction On souhaite réaliser un séquenceur à l'aide du micro-contrôleur STM32L476RG. Ce séquenceur devra être programmable à partir d'un PC relié au micro-contrôleur par une ligne série. Sa fréquence de fonctionnement sera par ailleurs réglable à l'aide de deux boutons-poussoirs. Ce séquenceur sera appliqué à la commande d'une série de leds (appelée couramment chenillard).



Le principe consiste simplement à allumer et éteindre une série de leds suivant un cycle particulier comme illustré sur le chronogramme suivant :



- Les leds commandées seront au nombre de 4 et reliées aux broches 12 à 15 de GPIOB.
- La longueur du cycle sera variable (8 périodes dans l'exemple), chaque période durant 240 ms (cette valeur pourra être modifiée par la suite).
- Un cycle sera stocké dans un tableau d'entiers dont la taille sera fixée par la longueur de la séquence (8 dans l'exemple).
- Les 4 bits de poids faible de chaque élément du tableau coderont l'état des leds correspondantes (0 pour éteint, 1 pour allumé). Par exemple pour allumer les leds D1 et D2 on utilisera la valeur 0110 en binaire soit 6 en décimal.
- L'ensemble du cycle sera stocké dans un tableau, par exemple pour le chronogramme précédent : `unsigned int cycle[8]={0, 8, 12, 4, 6, 2, 3, 1}`

1 - Séquencement de la commande

On choisit d'allumer ou d'éteindre les leds à la période fixée (240 ms) en utilisant l'interruption périodique provoquée par un timer de l'unité T2.

Écrire la fonction d'initialisation du timer et la fonction d'interruption périodique correspondante. Dans cette fonction on se contentera, dans un premier temps, d'allumer puis d'éteindre une led.

Ne pas oublier l'initialisation du port GPIOx pour piloter les leds.

2 - Exécution d'un cycle

Le cycle de fonctionnement du chenillard étant fixé et stocké dans un tableau, modifier le programme précédent pour que celui-ci exécute ce cycle en commandant les 4 leds.

3 - Téléchargement du cycle

Pour modifier le cycle à partir du PC on se propose d'écrire une fonction de réception sur la ligne série qui fonctionne par interruption (exécutée à chaque réception d'une donnée). Cette fonction permet de lire les N octets envoyés du PC vers le micro-contrôleur.

Le protocole de communication suivant est choisi :

- La transmission s'effectue à 19200 bauds avec des mots de 8 bits de donnée, sans bit de parité et avec un seul bit de stop.
- Le premier octet reçu représente la longueur L du cycle (nombre de périodes).
- Le début de la transmission (réception du premier octet) doit entraîner l'arrêt du chenillard (i.e. du timer).
- Les octets suivants représentent la valeur des leds pour une période donnée en commençant par la période 0. Les valeurs des 4 leds (0 : éteinte, 1 : allumée) sont codées dans les 4 bits de poids faibles de l'octet transmis.

- Après réception des N octets, le chenillard doit repartir à partir de la première période du nouveau cycle.

A titre d'exemple, le cycle illustré par le chronogramme correspond à la réception consécutive des valeurs suivantes : 8, 0, 8, 12, 4, 6, 2, 3, 1 (le premier chiffre est la longueur L du cycle).

Modifier le programme précédent afin de pouvoir télécharger le cycle envoyé sur la ligne série à partir du PC, et ensuite l'exécuter.

Remarques Remarques : La longueur du cycle pouvant varier, on utilisera un tableau statique de dimension suffisante (20 par exemple). L'émission lancée dans une fenêtre du PC permet d'envoyer des octets sur la ligne série.

4 - Réglage de la fréquence

On souhaite modifier la période de base du chenillard en utilisant deux boutons-poussoirs :

- Un bouton permettra de doubler la période à chaque appui (avec une période maximale de 960 ms),
- l'autre bouton divisera par deux la période avec une période minimale de 60 ms.

On utilisera pour cela deux entrées d'interruption externes.

1. Mettre en place, à partir du programme précédent, les deux interruptions et vérifier (en mettant des points d'arrêt) qu'elles fonctionnent.
2. Faire la modification de période dans les fonctions d'interruptions. Tester.

Travaux Pratiques 2

Conversion Analogique-Numérique Nous souhaitons réaliser un voltmètre utilisant l'unité ADC du STM32L476RG. Il accepte des signaux analogiques compris entre 0 et 3.3V en entrée. Ce voltmètre est connecté à un PC distant par une liaison série afin de pouvoir émettre des valeurs sur demande.

1 - Conversion simple

On souhaite lire une tension en mode conversion unique sur l'entrée analogique 1 de l'unité ADC.

Écrire votre programme et vérifier la pertinence du résultat avec le débogueur : prendre connaissance de la valeur lue sur un point d'arrêt ou bien faire une exécution pas à pas.

2 - Voltmètre

La conversion de la tension présente sur l'entrée 1 du convertisseur étant faite en permanence, on souhaite allumer des leds reliées au STM32 suivant la valeur de cette tension.

Pour une tension comprise :

- entre 0 et 0.6V, toutes les leds seront éteintes,
- entre 0.6V et 1.2V on allumera 1 led,
- entre 1.2V et 1.8V on allumera 2 leds,
- entre 1.8V et 2.4V on allumera 3 leds,
- entre 2.4V et 3.3V on allumera 4 leds.

Ecrire et mettre au point le programme permettant ce fonctionnement.

3 - Communication avec le PC via la ligne série

On souhaite envoyer la valeur de la tension d'une entrée du convertisseur vers le PC sur requête de celui-ci, requête envoyée du PC et reçue elle même sur la ligne série du STM32 (La ligne série du STM32 est reliée à la ligne série du PC).

La transmission s'effectue en asynchrone à 19200 bauds avec des mots de 8 bits sans bit de parité et avec un seul bit de stop. Les erreurs éventuelles de transmission ne sont pas gérées.

Initialisation de l'unité série USART3

Vérifier le bon fonctionnement de l'initialisation en écrivant un petit programme qui envoie un ou plusieurs caractères sur la ligne série du STM32 à destination du PC.

Réception de requêtes du PC vers le STM32

Mettre en œuvre une interruption sur la réception d'une donnée sur la ligne série. Cette donnée (envoyée à partir du PC) contient le numéro de l'entrée analogique sur laquelle on souhaite faire une conversion.

Émission vers le PC de la réponse à la requête

Compléter la fonction d'interruption de la question précédente de manière à :

- effectuer la conversion sur l'entrée choisie,
- la convertir en dixièmes de volts,
- la transmettre sur la ligne série (on vérifiera que le PC reçoit la bonne valeur).

Remarque : On choisi de calculer les valeurs de tension en dixièmes de volts (33 par exemple pour 3.3V) pour n'avoir à envoyer qu'une donnée, codable sur 8 bits.