

Série d'exercices 2 : Fonctionnalités avancées des microcontrôleurs

Exercice 1 : Interruption externe (Rising, Falling, Change)

Objectif : Utiliser `attachInterrupt()` pour réagir à un événement externe.

Monter un système où :

- Un bouton sur INT0 (rising edge) allume une LED
- Un bouton sur INT1 (falling edge) éteint la LED
- Un capteur sur un pin change interrupt (CHANGE) bascule l'état d'une deuxième LED

Afficher l'état sur le moniteur série à chaque interruption.

Exercice 2 : Conversion analogique-numérique par interruption

Objectif : Lire un potentiomètre sans blocage.

- Configurer l'ADC en mode interruption
- À chaque fin de conversion, stocker la valeur et déclencher une action
- Afficher la valeur sur LCD
- Si la valeur dépasse un seuil, allumer une LED d'alerte

Exercice 3 : Interruptions des timers (Timer0, Timer1, Timer2)

Objectif : Générer des événements périodiques précis.

- Timer0 : générer une interruption toutes les 1 ms pour incrémenter un compteur global
- Timer1 : toutes les 500 ms, alterner l'état d'une LED
- Timer2 : toutes les 2 secondes, envoyer un message série "Tic"

Afficher le compteur ms sur LCD 2x16.

Exercice 4 : PWM pour servo-moteur et variateur DC (L298N)

Objectif : Commander un servo et un moteur DC.

Partie A : Servo-moteur

- Utiliser le mode OC1A pour générer un signal PWM 50 Hz
- Balayer de 0° à 180° puis retour (période 5s)

Partie B : Moteur DC avec L298N

- Deux boutons : vitesse + / -
- Afficher le rapport cyclique sur TM1637
- Utiliser Timer2 en mode PWM rapide

Exercice 5 : Gradateur de puissance AC avec SSR et encodeur incrémental

Objectif : Contrôler une charge AC et naviguer dans un menu.

- Lire un encodeur rotatif (interruptions sur les canaux A et B)
- Afficher un menu sur LCD : "Puissance : XX%"
- Augmenter/diminuer le rapport cyclique avec l'encodeur
- Commander un SSR (sortie numérique) avec détection de passage par zéro

Exercice 6 : Communication UART (String, int, double) et stockage EEPROM

Objectif : Échanger des données typées et les sauvegarder.

1. Recevoir via UART des commandes :

- "SET X" où X est un entier → mémoriser en EEPROM
- "GET" → renvoyer la valeur stockée
- "CLEAR" → effacer l'EEPROM

Université Djilali BOUNAAMA, Khemis Miliana
مليانة خميس بونعامة جيلالي جامعة
Faculté des Sciences et de la Technologie
والتكنولوجيا العلوم كلية



2. Gérer les erreurs de parsing
3. Tout afficher sur le moniteur série