

UNIVERSITÉ MOHAMED KHIDER BISKRA
FACULTÉ DES SCIENCES ET DE LA TECHNOLOGIE
1^{ÈRE} ANNÉE MASTER AUTOMATIQUE ET INFORMATIQUE INDUSTRIELLE
SYSTÈMES EMBARQUÉS ET SYSTÈMES TEMPS RÉEL

SÉRIE N° 4

Exercice 1.

1. Proposer une version personnalisée de la fonction `delay()` de l'Arduino en utilisant le timer 0 en mode normal.
2. Utiliser la fonction proposée pour faire clignoter une LED avec une fréquence égale à 1 Hz.

Exercice 2. En utilisant le timer 0 en mode normal, écrivez un programme qui permet de faire clignoter une LED avec une fréquence égale à 1 Hz en assurant que le microcontrôleur ne soit pas bloqué.

Exercice 3. On veut réaliser un régulateur de vitesse d'un moteur à courant continu de faible puissance à base d'une carte Arduino Uno. La mesure de la vitesse de rotation du moteur est réalisée avec un codeur incrémental solidaire à son arbre de rotation. Le codeur est constitué d'un disque, ayant 200 trous équidistants, qui tourne entre une photodiode D et un phototransistor T. Le capteur délivre une tension périodique dont la fréquence est proportionnelle à la vitesse de rotation du moteur. La commande du moteur est réalisée en utilisant un pont en H commandé par la carte Arduino uno.

- Réaliser le montage du système de commande et écrire un programme réalisant une régulation par un correcteur proportionnel.
- Écrire un programme réalisant une régulation par un correcteur PI.
- Écrire un programme réalisant une régulation par un correcteur PID.

Exercice 4.

- Proposer une version personnalisée de la fonction `delay()` de l'Arduino en utilisant le timer 0 en mode TCT.
- Utiliser la fonction proposée pour faire clignoter une LED avec une fréquence égale à 1 Hz.

```

const int ledPin = 5;
#define COUNTTOFFSET 6

void setup() {
  TCCR2A = 0;    //normal mode
  TCCR2B = 0x04; //64 prescale 4 micro-seconds
  pinMode(ledPin,OUTPUT);
}

void mydelay(int x){
  while(x)
  {

    TCNT2=COUNTTOFFSET; //count up from here to 255 then over flow
    //TIFR2=1<<TOV2; //clear flag, bit is set on overflow
    bitSet (TIFR2, TOV2);
    //while( !(TIFR2&(1<<TOV2))); //wait for overflow bit
    while(!bitRead(TIFR2, TOV2));
    x--;
  }
}

void loop() {
  digitalWrite(ledPin,HIGH);
  mydelay(500);
  digitalWrite(ledPin,LOW);
  mydelay(500);
}

```