

# Prosit 6

## Résumé du contexte :

L’équipe a continué d’avancer, ils ont réussi à configurer les capteurs mais ils rencontrent des problèmes : comment faire deux tâches en même temps ?

## Définition mots-clefs :

ALU

Protocole de communication SPI

Librairie software Serial

Port-série/ Port analogique

Horodaté

Registre

Interruption

## Problématique :

**Comment pouvoir exécuter deux actions simultanément ?**

## Contraintes :

Une seul ALU

Impossible d’utiliser le même interface série pour le GPS et pour la configuration du système

Montage du test

Respecter le programme

## Pistes de solutions :

Avoir deux exécutions différentes en simultanées

Remplacé if par while

Arrêt de récupération des données

Toutes les 15 secondes allumer une LED qui permet d’appuyer sur un bouton

Arrêt prévu de la récupération

Mettre un compteur pour savoir combien de temps on a appuyé sur le bouton

Faire une fonction qui appelle dans le loop

## Généralisation :

Savoir faire des interruptions avec Arduino

Comprendre les registres

Comprendre le fonctionnement du microprocesseur

## Livrables :

Code amélioré qui permet de faire les deux actions du montage en même temps.

Plan d’action :

Tester le montage avec le code

Modifier le code pour pourvoir allumer la LED à chaque fois avec les pistes de solution

## Réalisation du plan d’action :

<https://roboticsbackend.com/how-to-do-multitasking-with-arduino/>

J’ai donc après lecture de cette ressource réalisé le code suivant :

|  |
| --- |
| #define PUSHPIN 8  #define LEDPIN 7  #define BLINKLED 5  unsigned long previousTimeLed = millis();  int ledState1 = LOW;  void setup()  {  Serial.begin(9600);  pinMode(LEDPIN, OUTPUT); // Initialisation LED  pinMode(BLINKLED, OUTPUT); // Initialisation LED  pinMode(PUSHPIN, INPUT); // Initialisation bouton  }  void loop()  {  unsigned long currentTime = millis();  // task 1  if(currentTime - previousTimeLed > 1000) {  previousTimeLed = currentTime;  if (ledState1 == HIGH) {  Serial.print("HIGH");  ledState1 = LOW;  }  else {  ledState1 = HIGH;  }  digitalWrite(BLINKLED, ledState1);  if (digitalRead(PUSHPIN) == LOW) {  digitalWrite(LEDPIN, 1);  }  else {  digitalWrite(LEDPIN, 0);  }  }    } |

Associé au prototype suivant :

