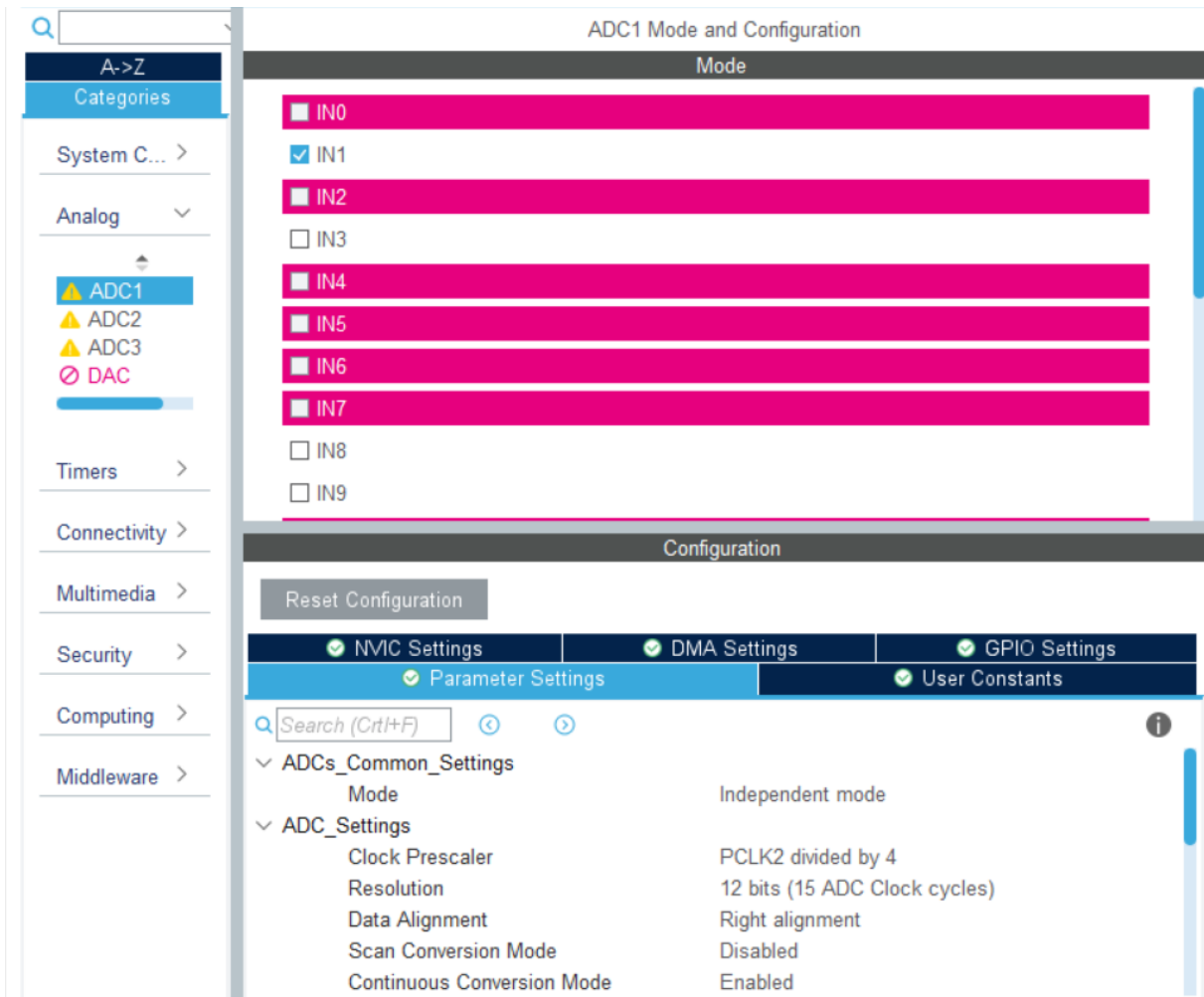


Ce tp conciste à afficher la valeur d'un potentiomètre sur une console putty

Pour ce faire on doit activer le ADC ainsi que l'USART :



Ce projet est diviser en trois version :

Version 1 :

```
while (1)
{
    HAL_ADC_Start(&hadc1);
    HAL_ADC_PollForConversion(&hadc1, 100);
    int adcData = HAL_ADC_GetValue(&hadc1);
    HAL_ADC_Stop(&hadc1);
    int duty = (adcData * 100)/4055;
    sprintf(message,"Valeur ADC: %d %%\n",duty);
    ecris_txt(message);
    HAL_Delay(200);
    /* USER CODE END WHILE */
    MX_USB_HOST_Process();

    /* USER CODE BEGIN 3 */
}
```

Ici on démarre l'ADC avec HAL_ADC_Start et HAL_ADC_PollForConversion, on récupère la valeur du potentiomètre avec HAL_ADC_GetValue, on la convertit en faisant un produit en croix (nous voulons un pourcentage ici), et enfin on envoie un message dans la console putty

Version 2 :

Dans cette version, si la valeur du potentiomètre ne bouge pas, elle ne se réaffiche pas en boucle sur le terminal :

```
int old_val = 0;
while (1)
{
    HAL_ADC_Start(&hadc1);
    HAL_ADC_PollForConversion(&hadc1, 100);
    int adcData = HAL_ADC_GetValue(&hadc1);
    HAL_ADC_Stop(&hadc1);
    int duty = (adcData * 100)/4055;
    if(old_val != duty){
        sprintf(message, "Valeur ADC: %d %%\n", duty);
        ecris_txt(message);
        old_val = duty;
    }

    HAL_Delay(200);
    /* USER CODE END WHILE */
    MX_USB_HOST_Process();
}
```

Pour cela, on rajoute une variable old_val qui est égale à la valeur précédente, si la valeur actuelle et la valeur précédente sont égales, le message ne s'affiche pas.

Version 3 :

Dans cette version on utilise la fonction proche qui nous permet d'avoir un intervalle moins sensible :

```
#define delta 10
int proche(uint32_t x, uint32_t y) {
    int dx=x-y;
    dx=(dx<0)?-dx:dx;
    return dx<delta;
}

int old_val = 0;
while (1)
{
    HAL_ADC_Start(&hadc1);
    HAL_ADC_PollForConversion(&hadc1, 100);
    int adcData = HAL_ADC_GetValue(&hadc1);
    HAL_ADC_Stop(&hadc1);
    int duty = (adcData * 100)/4095;
    float v = (adcData * 3.3)/4095;
    if(! proche(adcData, old_val)){
        sprintf(message, "Valeur ADC: %x -> pourc %d %% -> V = %.2f \n", adcData, duty, v);
        ecris_txt(message);
        old_val = adcData;
    }

    HAL_Delay(200);
    /* USER CODE END WHILE */
    MX_USB_HOST_Process();

    /* USER CODE BEGIN 3 */
}
/* USER CODE END 3 */
```

Cette version est similaire à la précédente, sauf que l'on utilise la fonction proche qui regarde un intervalle de 10, si la valeur actuelle est supérieur de 10 à la valeur précédente, le message est affiché.