

Table des matières

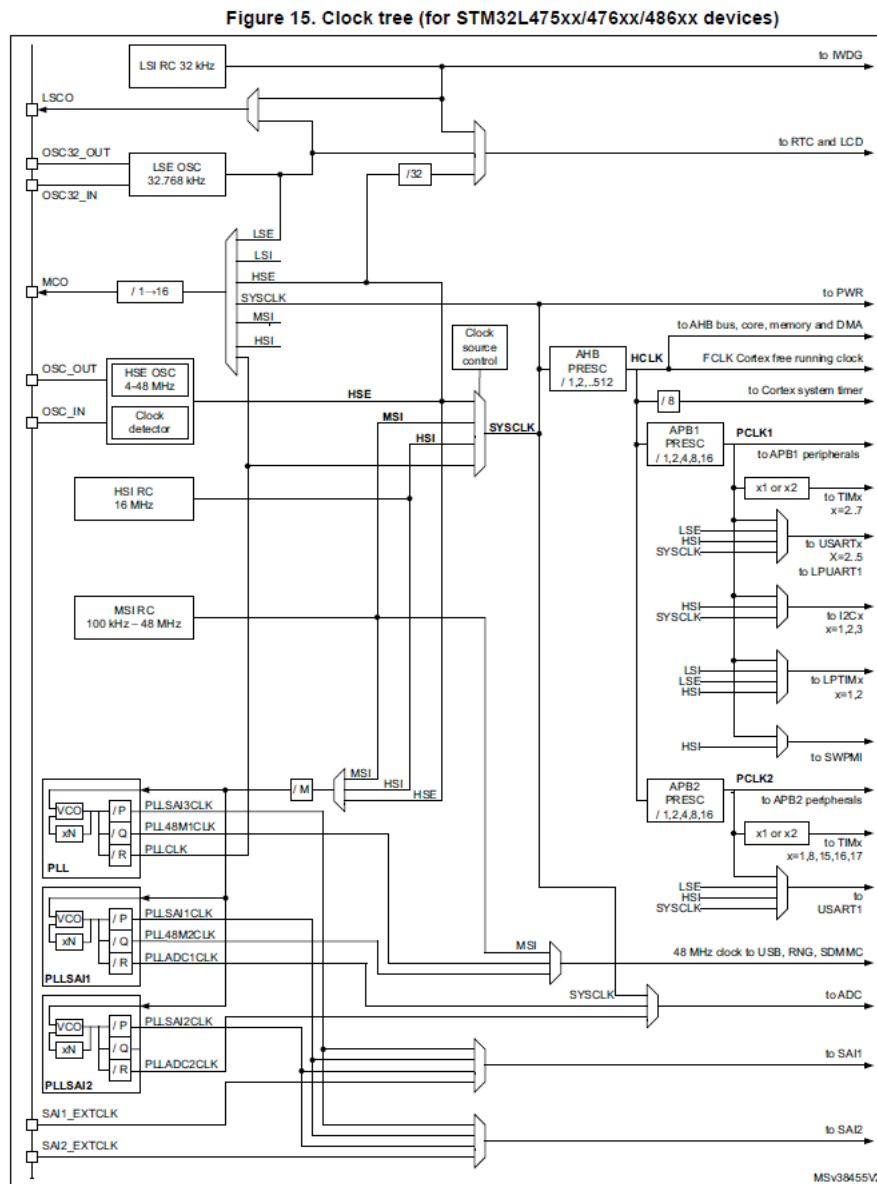
I) COMMENT REALISER UNE FONCTION DELAY μS	2
I-1) LANCER STM32CUBEMX.	2
I-1-A) PINOUT & CONFIGURATION :	2
I-1-B) ONGLET CLOCK CONFIGURATION :	3
I-1-C) PINOUT & CONFIGURATION :	3
I-2) LANCER KEIL μ VISION	3
II) TEST	3

l) Comment réaliser une fonction delay μs

l-1) Lancer STM32CubeMX.

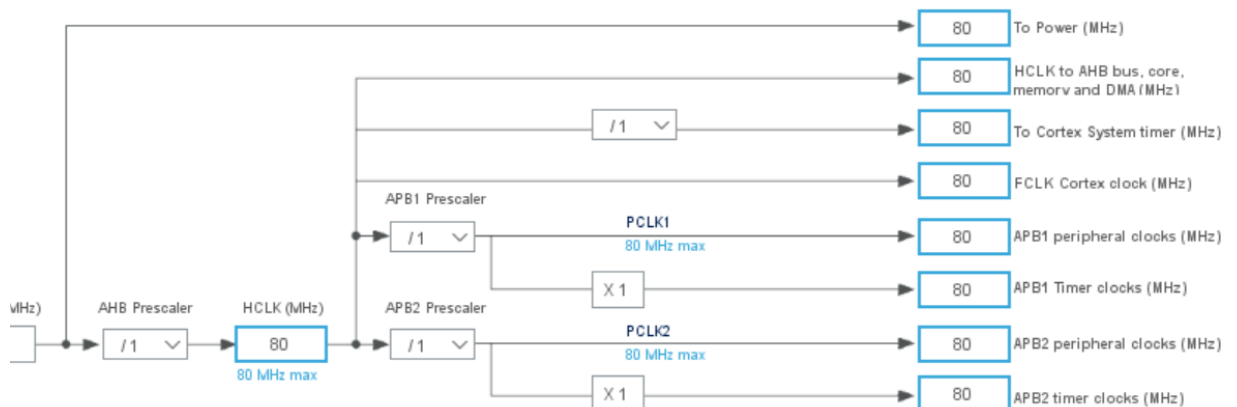
l-1-a) Pinout & Configuration :

- Regarder dans le datasheet sur quel APBx se trouve le timer que l'on souhaite utiliser
 - o Choisissons le timer 1



- o On constate que **TIM1** est sur la ligne d'horloge APB2.

I-1-b) Onglet Clock configuration :



- On voit que **APB2** est déjà réglé sur la fréquence maximum du STM32L476RG, ici **80MHz**.

○

I-1-c) Pinout & Configuration :

- Choisir **tim1**
 - Clock source → **internal clock**
 - **Parameter settings**
 - Prescaler → **80-1**
 - ARR → **0xffff -1**
 - **PinoutView**
 - Choisir PA1 pour visualisation à l'oscilloscope → **GPIO_OUTPUT**
 - **Générer le code**

I-2) Lancer Keil µVision

```
/* Private user code -----*/
/* USER CODE BEGIN 0 */
void delay (uint16_t delay) // pour obtenir un delai en microsecondes
{
    __HAL_TIM_SET_COUNTER (&htim1, 0);
    while(__HAL_TIM_GET_COUNTER(&htim1) < delay);
}
/* USER CODE END 0 */
```

II) Test

Il suffit maintenant d'appeler **delay(1)** à chaque fois que l'on a besoin d'un délai d'une microseconde.