Assignment 8

Create a RTOS Application
Create Task1 and Task2 with their proper TaskHandles.
Pass the following parameters:
Task 1 (handle)----Pass to ----> Task1 as parameter
Task 2 (handle)----Pass to ----> Task2 as parameter
Inside Task1:
Suspend Task 2 using its handle using proper API.
Blink LED1 and LED2 at every 500ms for 10 times, after that Resume Task 2 using proper API.
Inside Task2:
Blink LED3 and LED4 at every 100ms infinite times.

The task functions are defined as below:

```
void Task1(void *T2)
{
	TaskHandle_t *xHandle2 = (TaskHandle_t *)T2;
	vTaskSuspend(*xHandle2);
	uint16_t count = 10;
	for(;;)
	{
		HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_12 | GPIO_PIN_13);
		HAL_Delay(500);
		if(count == 1)
		{
			vTaskResume(*xHandle2);
		}
		count--;
		vTaskDelete(NULL);
	}
}

void Task2(void *T1)
{
	for(;;)
	{
		HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_14 | GPIO_PIN_15);
		HAL_Delay(100);

	}
}
```

The task 1 function is defined such that the led 1 and 2 toggle every 500ms, 10 times.

While, task 2 function is defined so that the led 3 and 4 toggle every 100ms infinite times.

vTaskSuspend():

Places a task into the Suspended state. A task that is in the Suspended state will never be selected to enter the Running state.
The only way of removing a task from the Suspended state is to make it the subject of a call to vTaskResume().

vTaskResume():

Transition a task from the Suspended state to the Ready state. The task must have previously been placed into the Suspended state using a call to vTaskSuspend().

The trace of the code is shown below: