

Prelab:

1) You will need to turn a GPIO pin on and off. From your last lab, provide the code that was used to turn a GPIO pin on and the code that was used to turn a GPIO pin off.

```
HAL_GPIO_TogglePin(LD_R_GPIO_Port, LD_R_Pin); // on
HAL_Delay(500);
HAL_GPIO_TogglePin(LD_R_GPIO_Port, LD_R_Pin); // off
HAL_Delay(500);
```

Note: The delays are not needed but it's realistic to have them as you won't be able to observe the lights turning off and on without them. They were also a part of my lab code.

2) Review the data sheet for the SK9822. Provide the 32-bit pattern for light purple and the 32-bit pattern for yellow. Provide the numbers in hexadecimal form. Explain how you calculated them.

Use e4 to get 111 and moderate luminance, then in pattern bgr, 00ffff is yellow and light purple would be E3C3CB (used google for purple).

Light purple:0xe4e3c3cb

Yellow:0xe400ffff

3) Identify where in the manuals the HAL_Delay() function API is documented.

It is found in "Description of STM32L4/L4+ HAL and low-layer drivers - User manual" on page 33/2719.

It states:

"HAL_Delay(). this function implements a delay (expressed in milliseconds) using the SysTick timer. Care must be taken when using HAL_Delay() since this function provides an accurate delay (expressed in milliseconds) based on a variable incremented in SysTick ISR. This means that if HAL_Delay() is called from a peripheral ISR, then the SysTick interrupt must have highest priority (numerically lower) than the peripheral interrupt, otherwise the caller ISR will be blocked."