

## Bloque 3

### RESUMEN

El **BLOQUE 3 (USO DE CAPAS DE ABSTRACCIÓN)** se divide en los siguientes 4 temas. Todos los temas de este Bloque tienen videos explicativos, donde no sólo se explica la teoría, sino también los ejemplos ofrecidos.

---

En el **tema 9 (Trabajando con bibliotecas: HAL)** se enseña cómo utilizar la abstracción hardware, tanto a nivel visual, como utilizando bibliotecas proporcionadas por el fabricante del microcontrolador. Esto se explicará siguiendo la siguiente secuencia de apartados:

En el primer punto se habla de la abstracción hardware, con los conceptos generales, los pros, los contras, y los tipos de abstracción que se van a tratar en el curso.

A continuación se describe la abstracción visual, ya que es habitual que los fabricantes proporcionen herramientas que ayuden al desarrollador a elegir los recursos a utilizar. En el caso de ST Microelectronics, se proporciona la perspectiva CubeMX como parte del sistema STM32CubeIDE.

En el tercer punto se explica la abstracción hardware a nivel software, mediante el uso de las *Hardware Abstraction Libraries* (HAL). ST Microelectronics proporciona, además de este tipo de bibliotecas, otras de más alto nivel, específicas para incluir servicios de cada una de las placas de desarrollo que venden (por ejemplo, la STM32L-DISCOVERY).

En el cuarto punto se muestra un ejemplo completo del uso de HAL y la perspectiva CubeMX, sin utilizar interrupciones.

En el último punto se muestra el mismo ejemplo, pero esta vez utilizando interrupciones.

En el **tema 10 (Comunicación Serie Asíncrona: USART)** se enseña cómo utilizar el bloque USART del microcontrolador. Esto se hace a través de la siguiente estructura:

En el primer punto se habla de los conceptos generales de las comunicaciones particularizando en la comunicación serie asíncrona.

En el segundo punto se describe la **arquitectura** del **BLOQUE USART**, indicando primero sus características básicas, después explicando cómo se configura en la perspectiva CubeMX y finalmente las funciones HAL que se utilizan para la programación.

En el tercer punto se explica cómo realizar la conexión física entre la placa STM32L-DISCOVERY y un PC utilizando puertos USB y módulos de conversión TTL-USB, así como lo que se debe configurar en el programa terminal del PC (por ejemplo, el Tera Term) para realizar la comunicación.

En el cuarto punto se muestran tres ejemplos prácticos con HAL y perspectiva CubeMX (el primero de transmisión por espera activa, el segundo de transmisión y recepción por espera activa y el tercero de transmisión por espera activa con recepción por interrupción). Estos ejemplos deben ser probados por el alumno, y se proporcionan los programas correspondientes (inicialización, funcionamiento continuo y variables globales y callbacks para el caso de las interrupciones).

En el quinto punto se describe cómo trabajar con la comunicación serie asíncrona, pero configurando y programando utilizando registros. Esto se proporciona sólo a título informativo, por si el alumno quiere comprobar lo que realmente llega a simplificar el desarrollo utilizar la abstracción hardware.

Los diferentes **registros de control** del bloque USART son los siguientes:

- USARTx→CR1
- USARTx→CR2
- USARTx→BRR

El único **registro de datos** del bloque USART, que sirve para leer/escribir los datos en los pines del bloque, es el siguiente:

- USARTx→DR.

El único **registro de estado** del bloque USART es el siguiente:

- USARTx→SR

En el último punto se muestran los tres mismos ejemplos prácticos que en el cuarto punto, pero ahora utilizando la configuración y programación directamente con registros.

---

En el **tema 11** (**Comunicación Serie Síncrona: I2C, SPI**) se enseñan los fundamentos teóricos de la comunicación serie síncrona, particularizado en los dos protocolos más comúnmente usados.

Primeramente, se estudian de los conceptos generales de la comunicación serie síncrona.

A continuación, se describe el protocolo **I2C** y finalmente se describe el protocolo **SPI**.

---

En el **tema 12** (**Diseño de Soluciones y Funciones Especiales: RTC, Watchdog y Consumo**) se enseña cómo diseñar un proyecto real y algunas funciones especiales interesantes del microcontrolador, dividido en varios apartados:

El tema se inicia con la explicación de la funcionalidad de reloj en tiempo real (RTC), proporcionando un ejemplo con HAL y perspectiva CubeMX para probar por el alumno, con su programa correspondiente (inicialización, y funcionamiento continuo). Después del ejemplo, tal y como se hizo con la USART, se muestra cómo configurar el RTC con registros.

Los diferentes **registros de control** del bloque RTC son los siguientes:

- RTC→CR
- RTC→PRER
- RTC→WPR

Los dos **registros de datos** del bloque RTC son los siguientes:

- RTC→TR
- RTC→DR

El único **registro de estado** del bloque RTC es el siguiente:

- RTC→ISR

Una vez visto el RTC, se explica la funcionalidad de Watchdog y los conceptos de bajo consumo.

Posteriormente se explican los pasos a seguir a la hora de diseñar un proyecto completo y qué puntos hay que tener en cuenta en cuanto a temporización, recursos comunes y desarrollo.

En el último punto se muestra la utilización de las APIs en la programación, ampliando lo visto en el Tema 9.