Introduction to HAL library and LL library

1、STM32 library overview

1.1 Introduction to the standard library

The Standard Peripheral Library is the most basic library provided by STMicroelectronics. It provides direct access to the low-level registers and peripherals of the STM32 microcontroller. The standard library is designed to provide a high degree of flexibility and low-level hardware control to meet stringent performance and resource requirements. Using the standard library, developers can directly manipulate registers to configure and control the functionality of the microcontroller, but this requires manually writing a large amount of low-level code. The standard library is suitable for applications with higher performance requirements and projects that have strict requirements on code size and efficiency.

1.2. Features of the standard library

advantage:

(1) Portable: The standard library functions are suitable for all C language compilers and can be used in different development environments.

(2) Good compatibility: The standard library functions have good compatibility on different platforms and can be easily transplanted and debugged.

(3) Easy to learn and use: The calling method of standard library functions is similar to the function calling method of C language, making it easy to learn and use.

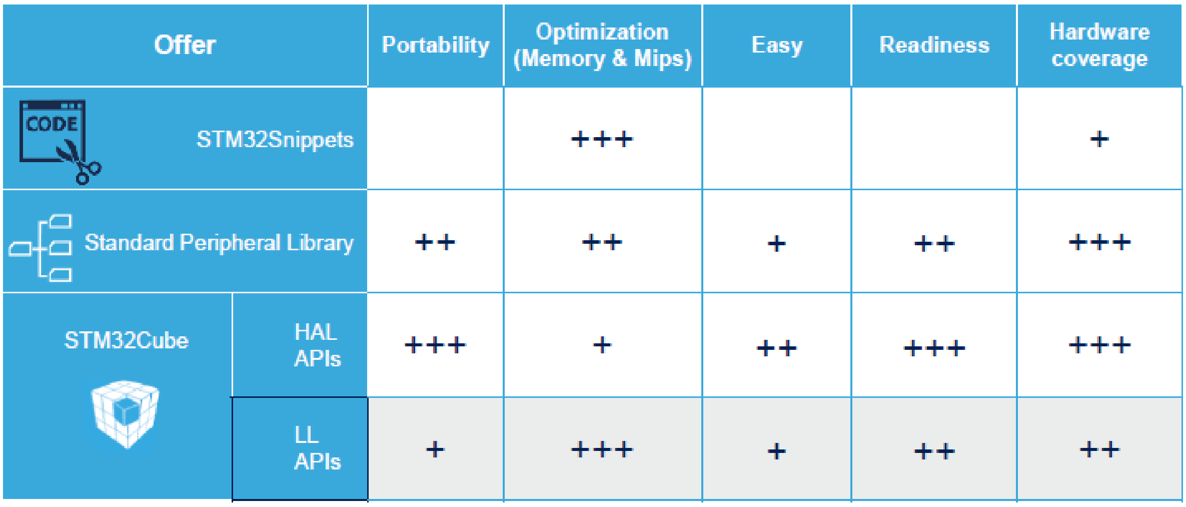
shortcoming:

(1) Low execution efficiency: The standard library functions have low execution efficiency and are suitable for some application scenarios that do not require high performance.

(2) The functions are relatively simple: The functions provided by the standard library functions are relatively simple and cannot meet some high-level needs.

(3) Hardware abstraction layer is not supported: The standard library functions cannot directly operate the hardware, and the hardware abstraction layer needs to be implemented through the underlying driver.

The comparison between the standard library, HAL library and LL library is as shown below



Offer		Portability	Optimization (Memory & Mips)	Easy	Readiness	Hardware coverage
STM32Snippets			+++			+
Standard Peripheral Library		++	++	+	++	+++
STM32Cube	HAL APIs	+++	+	++	+++	+++
	LL APIs	+	+++	+	++	++

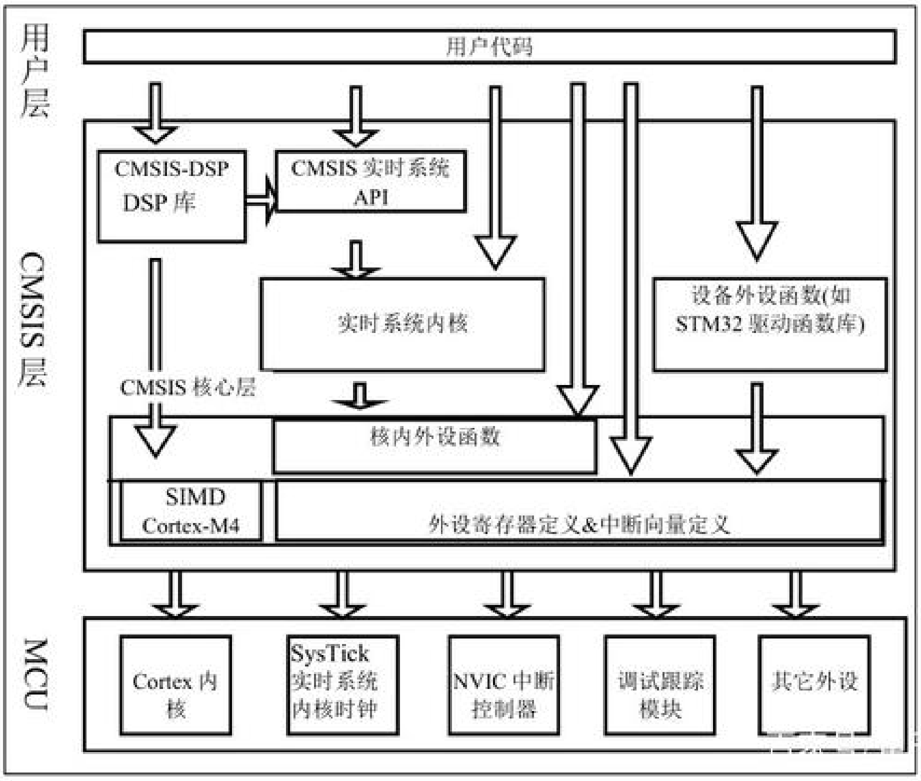
Each of the three libraries has its own advantages and disadvantages. **However, HAL and LL are currently the official libraries that are mainly recommended and maintained and updated. So we will focus on the HAL library and LL library.**

2. HAL library

2.1 Introduction to HAL library

HAL is the abbreviation of Hardware Abstraction Layer, Chinese name: Hardware Abstraction Layer.

The HAL library is an abstraction layer embedded software launched by ST for STM32, which can better ensure the maximum portability of STM32 products. The library provides a consistent set of middleware components such as RTOS, USB, TCP/IP and graphics. The HAL library is open source code released under a non-restrictive BSD license (Berkeley Software Distribution). The middleware stack (USB Host and Device Library, STemWin) produced by ST comes with a licensing model that allows easy reuse, as long as it is used on ST's MCU chips. The middleware in the library (USB Host/Device Library, STemWin) protocol The stack is allowed to be modified at will and can be used repeatedly.
All in all, the HAL library allows users to use code across chips, and different MCU chips (F1/F2...) can be used.



2.2 Characteristics of HAL library

advantage:

(1) Easy to learn and use: The calling method of HAL library functions is similar to calling standard library functions, which is easy to learn and use.

(2) Support hardware abstraction layer: HAL library function provides a hardware abstraction layer, which can directly operate the hardware without the need for underlying drivers.

(3) Provides high-level API functions: HAL library functions provide a series of high-level API functions, including interrupt processing, DMA transmission and other functions, to facilitate rapid development by users.

(4) Portability: HAL library functions have good portability and can be used in different development environments.

shortcoming:

(1) Low execution efficiency: The execution efficiency of HAL library functions is low and is suitable for application scenarios that do not have high performance requirements.

(2) The function is relatively simple: The function of the HAL library function is relatively simple and cannot meet some high-level needs.

(3) Large space occupied: HAL library functions occupy large space and are suitable for application scenarios with sufficient resources.

3. LL library

The LL library (Low-Level Library) is a lower-level library provided by STMicroelectronics based on the HAL library.

The LL library provides more direct access to low-level registers and peripherals and provides a set of low-level API functions. The LL library retains more hardware details, providing developers with a higher level of flexibility and control.

Using the LL library, developers can directly write lower-level code to achieve fine control of microcontrollers and peripherals.

3.1 Characteristics of LL library

advantage:

(1) High execution efficiency: The interaction between LL library functions and hardware is more direct, the execution efficiency is higher, and it is suitable for application scenarios with high performance requirements.

(2) Closer to hardware: LL library functions are closer to hardware, providing higher flexibility and customizability, and are suitable for some application scenarios that require direct control of hardware, such as device drivers.

(3) Support hardware abstraction layer: LL library functions provide a hardware abstraction layer that can directly operate hardware without the need for underlying drivers.

shortcoming:

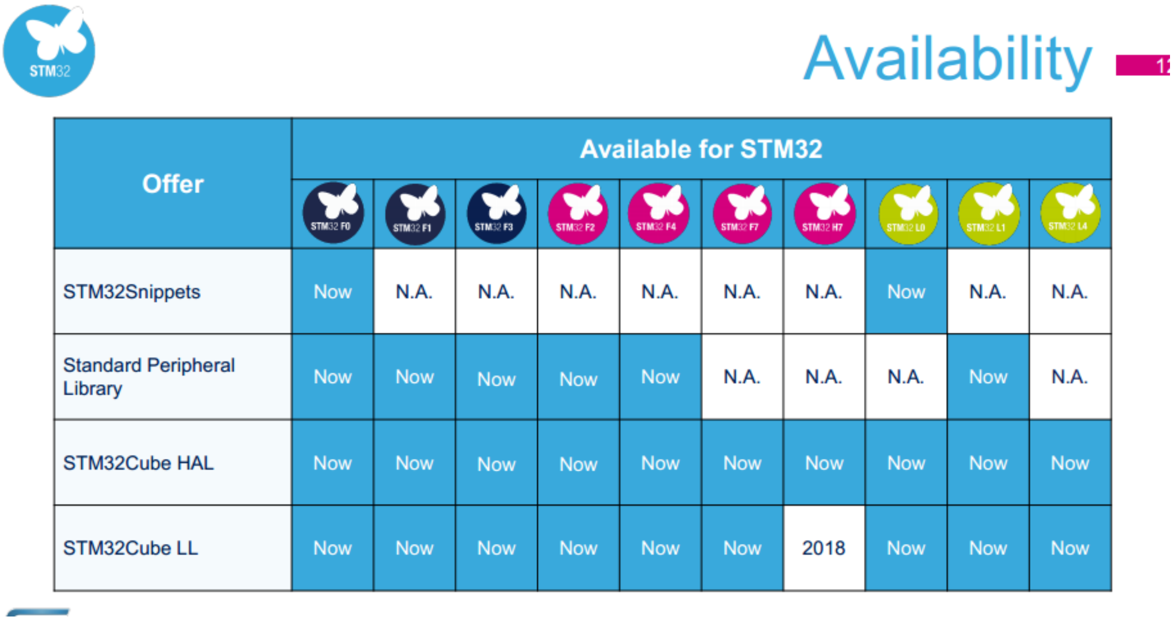
(1) Complex use: The use of LL library functions is relatively complex and requires a certain understanding of hardware control methods, so it is not suitable for beginners.

(2) Not portable: The portability of LL library functions is poor. Different models of STM32 chips may have different versions of LL library functions, which need to be selected according to the chip model.

(3) Does not support high-level functions: LL library functions do not support high-level functions, such as interrupt processing, DMA transmission, etc., and need to be implemented through other methods.

4. Comparison between HAL library and LL library

The support of stm32 by HAL library and LL library is as follows:



Availability

Offer	Available for STM32									
	STM32 F0	STM32 F1	STM32 F3	STM32 F2	STM32 F4	STM32 F7	STM32 H7	STM32 L0	STM32 L1	STM32 L4
STM32Snippets	Now	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	Now	N.A.	N.A.
Standard Peripheral Library	Now	Now	Now	Now	Now	N.A.	N.A.	N.A.	Now	N.A.
STM32Cube HAL	Now	Now	Now	Now	Now	Now	Now	Now	Now	Now
STM32Cube LL	Now	Now	Now	Now	Now	Now	2018	Now	Now	Now

Both the LL library and the HAL library have their own advantages and disadvantages, and developers need to choose which library to use based on specific application scenarios. The following is a comparison between the LL library and the HAL library:

- Application scenarios: LL library is suitable for application scenarios with high performance requirements, such as embedded systems, real-time control systems, etc.; HAL library is suitable for rapid development and prototype design, such as learning to develop STM32, rapid development of small projects, etc.
- Programming difficulty: LL library requires developers to directly operate registers, making programming more difficult; HAL library provides a higher level of abstraction, making programming less difficult.
- Code size: The LL library has less code and takes up less memory; the HAL library has more code and takes up more memory.
- Speed: The LL library is faster and suitable for applications with high performance requirements; the HAL library is slower and suitable for applications with low performance requirements.

In general, the LL library and the HAL library each have their own advantages and disadvantages, and developers need to choose which library to use based on specific application scenarios.