

CH32V20x Evaluation Board Reference

Version: V1.4

<https://wch-ic.com>

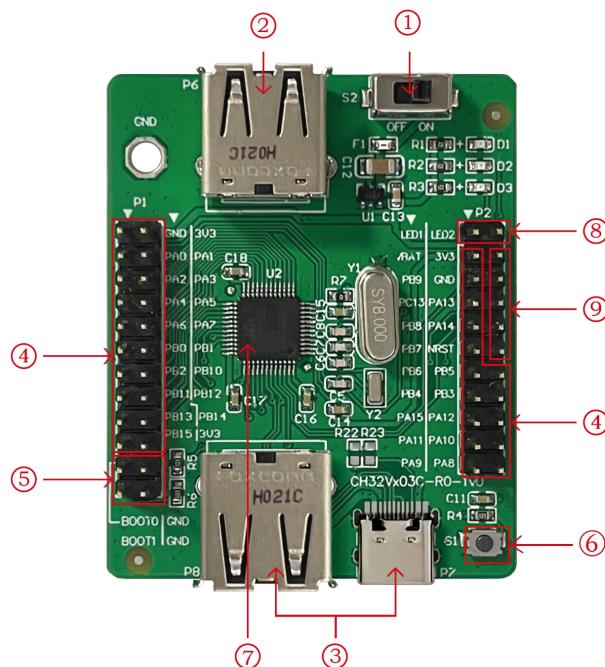
1. Overview

This evaluation board is applied to the development of the CH32V20x chip. The IDE uses the MounRiver compiler, with the option of using the on-board or independent WCH-Link for emulation and download, and provides reference examples and demonstrations of chip resource-related applications.

2. Evaluation board hardware

Please refer to the CH32V20xSCH.pdf document for the schematic of the evaluation board.

CH32V203 Evaluation Board



Descriptions

- | | | |
|------------------|----------------------------|-----------------|
| 1. Power switch | 4. MCU I/O port | 7.MCU |
| 2. USB interface | 5. Boot mode configuration | 8.LED |
| 3. USB interface | 6. Reset button | 9.SWD interface |

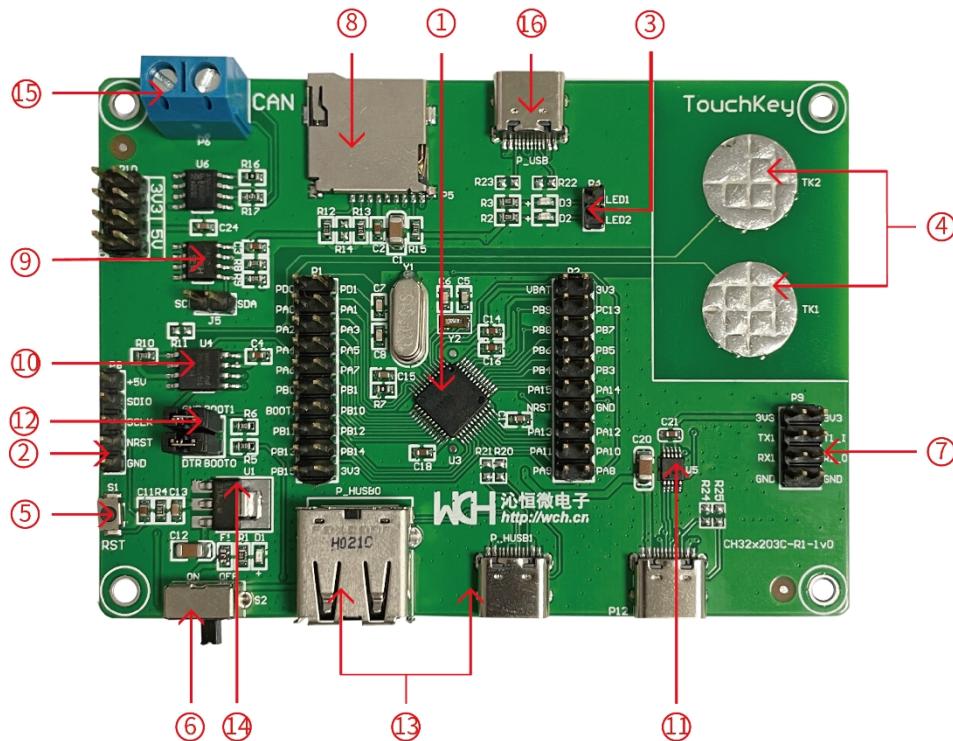
The above CH32V203 evaluation board comes with the following resources.

Motherboard - CH32V203EVT

1. Switch S2: Used to disconnect or connect external 5V power supply or USB power supply
2. USB interface P6: USB communication interface PB6, PB7 of the main chip
3. USB interface P7, P8: USB communication interface PA11, PA12 of the main chip
4. MCU I/O port P1, P2: I/O pinout interface of the main control MCU
5. Boot mode configuration P3: Select the boot mode when the chip is powered on by configuring BOOT0/1
6. Button S1: Reset button for external manual reset of the main control MCU
7. Main control MCU: CH32V203C8T6/ CH32V203C6T6
8. LED: Connected to the main chip I/O port via pins for control

9. Debug interface: For downloading, simulation debugging

CH32V203 Evaluation Board



Descriptions

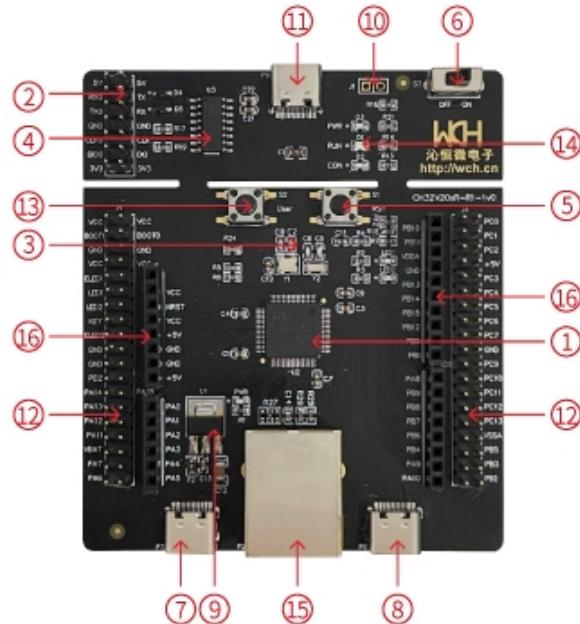
1.Main control MCU	5.Reset button	9.EEPROM chip U2	13.USB main-slave interface P_HUSB
2.SWD interface	6.Power switch	10.Serial Flash memory U4	14.Forward low dropout voltage regulator chip U1
3.LED	7.Serial port 1	11.RS232 level conversion chip U5	15.CAN interface P6
4.Touch button	8.SD card holder P5	12.Boot mode configuration	16.USB slave interface P_USB

Motherboard - CH32V203C-R1

1. Main control MCU: CH32V203C8T6
2. SWD interface: For downloading, simulation debugging
3. LED: Connected to the main chip I/O port through P4 pins for control
4. Touch keys: Connect the main chip touch keys channel 0, channel 1
5. Reset button: For external manual reset power supply switch
6. Power switch: Used to cut off or connect external 5V power supply or USB power supply
7. Serial port 1: Connect to the main chip URAT1 interface to demonstrate the serial port transceiver function
8. SD card holder P5: Connect to SPI1 interface to demonstrate the operation of TF card through SPI interface
9. EEPROM chip U2: Connects to I2C interface and connects to I/O of main chip through J5
10. Serial Flash memory U4: Connect SPI1 interface to demonstrate the operation of Flash memory
11. RS232 level conversion chip U5: Used to convert TTL signal of serial port to RS232 signal
12. Boot mode configuration: Select the boot mode when the chip is powered on by configuring BOOT0/1
13. USB interface P_HUSB: USB communication interface of the main chip, with Host and Device functions

14. Forward low dropout voltage regulator chip U1: Used to realize the conversion of 5V voltage to 3.3V supply voltage available to the chip
15. CAN interface P6: Connects to the main chip via CAN chip U6
16. USB interface P_USB: USB communication interface of the main chip, only Device function

CH32V203 Evaluation Board



Descriptions

1.Main control MCU	5.Reset button	9. Voltage regulator chip	13.User button
2.SDI&UART interface	6.Power switch	10.Download interface	14. WCH-Link LED
3.LED	7.USB2.0 full-speed interface	11.WCH-Link interface	15.Network port
4.WCH-Link MCU	8.USB2.0 high-speed interface	12.MCU I/O	16.AEDUINO interface

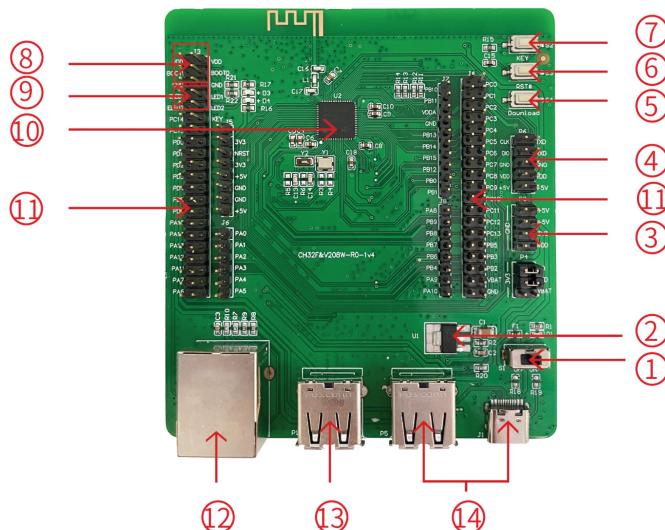
The CH32V203 evaluation board shown above comes with the following resources.

Motherboard - CH32V203EVT

1. Main control MCU: CH32V203RBT6
2. SDI & UART interface: For downloading, emulation debugging, need jumper to choose whether to use the on-board WCH-Link
3. LED: Connected to the I/O port of the main control MCU through J3 pins for control
4. WCH-Link MCU: MCU that implements WCH-Link function
5. Button S1: Reset button for external manual reset of the main control MCU
6. Switch S3: Used to cut off or connect external 5V power supply or USB power supply
7. USB type-C interface P7: Connect the main chip USB2.0 full-speed communication interface
8. USB interface P6: Connect the main chip USB2.0 high-speed communication interface
9. Voltage regulator chip U1: Used to realize the conversion of 5V voltage to 3.3V supply voltage available to the chip
10. Download interface J1: When J1 jumper is shorted, it can be used to achieve WCH-Link firmware update
11. WCH-Link interface: For connecting PC and WCH-Link function module
12. MCU I/O port: I/O pinout interface of main control MCU
13. USER button S2: Connect the I/O port of the main control MCU through J3 pin for key control

14. WCH-Link indicator: Including D1, D2 and D3 three LEDs, indicating the WCH-Link operation status
15. Network port: The network communication interface of the main chip
16. ARDUINO interface: Convenient to connect the ARDUINO interface development board

CH32V208 Evaluation Board



Descriptions

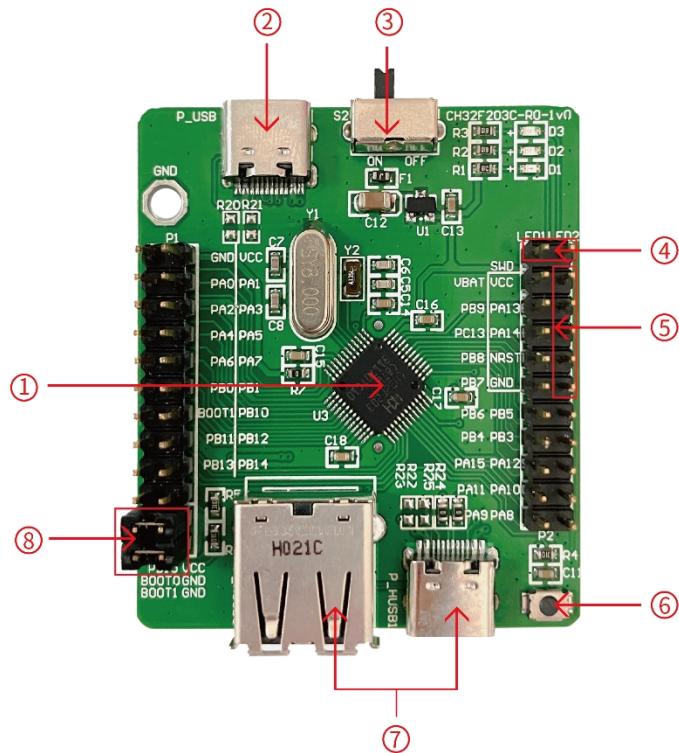
1.Power switch	5.Download button	9.LED row pin	13.USB interface
2.Voltage regulator chip	6.Reset button	10. Main control MCU	14.USB interface
3.Power row pin	7.KEY	11.MCU I/O port	
4.DEBUG interface	8.Boot mode configuration	12.Network port	

The CH32V208 evaluation board shown above comes with the following resources.

Motherboard - CH32V208EVT

1. Switch S1: Used to cut off or connect external 5V power supply or USB power supply
2. Forward low dropout voltage regulator chip U1: Used to realize the conversion of 5V voltage to 3.3V supply voltage available to the chip
3. Power supply pin P3: 5V, 3.3V, GND external power supply pin
4. DEBUG interface P6: For downloading, simulation debugging
5. Key S4: Download key, used to start download from BOOT
6. Button S3: Reset button for external manual reset of the main MCU
7. Key S2: Connect to the I/O port of the main MCU for key control through the P1 row of pins
8. Boot mode configuration: Select the boot mode when the chip is powered on by configuring BOOT0/1
9. The row of pins connects to the I/O of the main control MCU to control the LED
10. Main control MCU: CH32V208WBU6
11. MCU I/O port: I/O pinout interface of the main control MCU
12. Network port: Network communication interface of the main chip
13. USB interface P5, P15: USB communication interface PA11, PA12 of the main chip
14. USB interface P4, P14: USB communication interface PB6, PB7 of the main chip

CH32V203 Evaluation Board



Descriptions

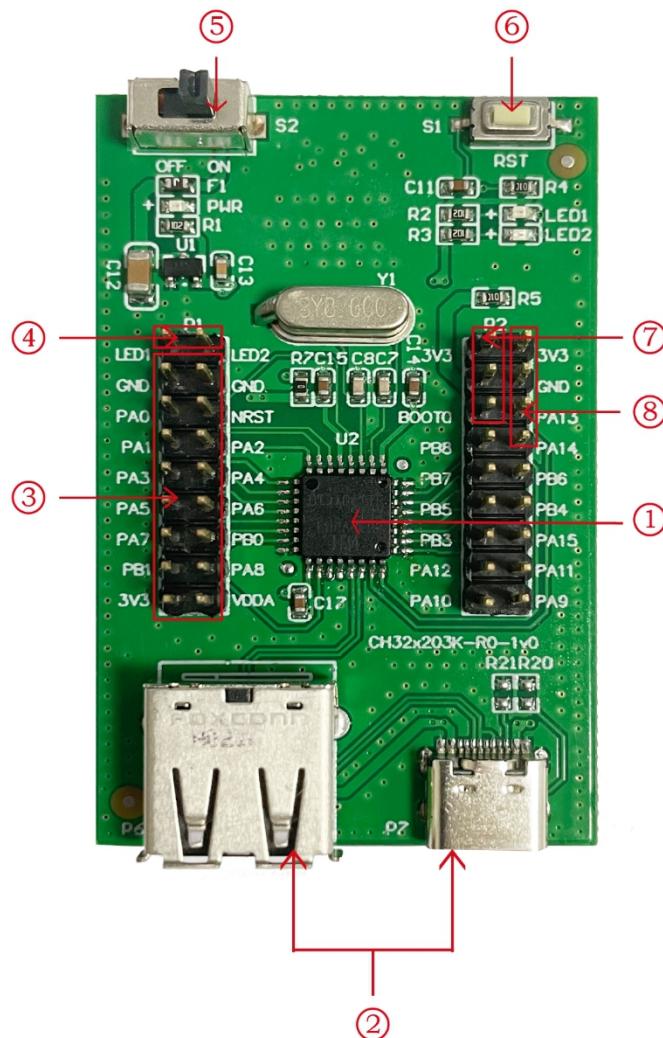
- | | | | |
|--------------------|-----------------------------|------------------------------|---------------------------|
| 1.Main control MCU | 2.USB slave interface P_USB | 3.Power switch | 4.LED |
| 5.SWD interface | 6.Reset button | 7.USB master-slave interface | 8.Boot mode configuration |

The CH32V203C-R0 EVT board comes with the following resources.

Motherboard - CH32V203C-R0

1. Main control MCU: CH32V203C8U6
2. USB interface P_USB: USB communication interface of the main chip, only Device function
3. Power switch: Used to cut off or connect external 5V power supply or USB power supply
4. LED: Connected to the main chip I/O port for control
5. SWD interface: For downloading, simulation debugging
6. Reset button: For external manual reset power supply switch
7. USB interface P_HUSB: USB communication interface of the main chip with Host and Device functions
8. Boot mode configuration: Select the boot mode when the chip is powered on by configuring BOOT0/1

CH32V203 Evaluation Board



Descriptions

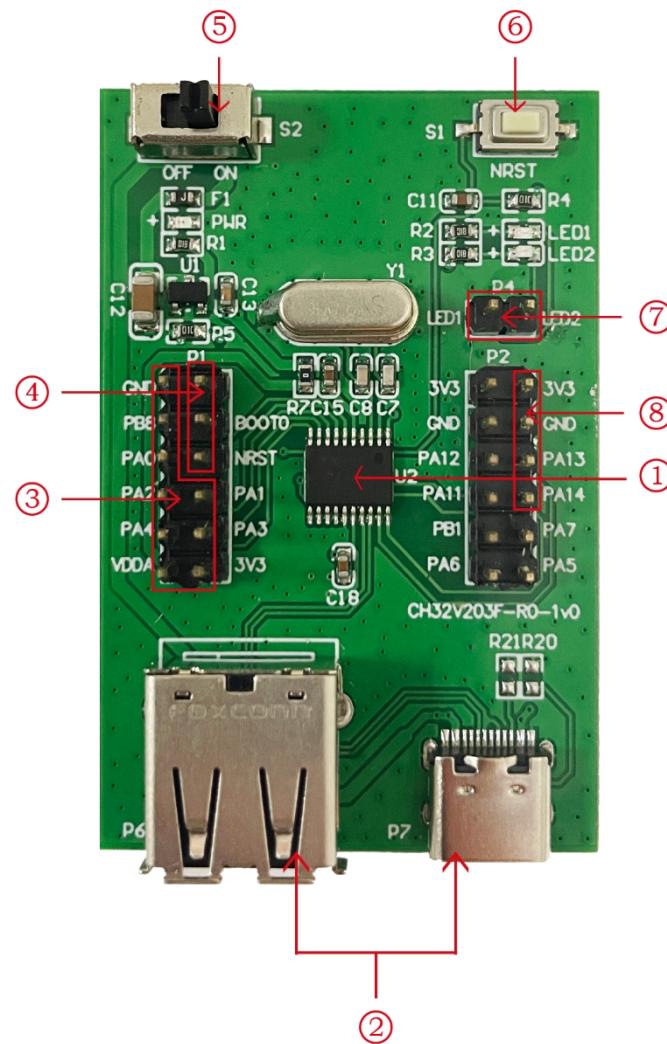
- | | | | |
|--------------------|-----------------|---------------------------|-----------------|
| 1.Main control MCU | 2.USB interface | 3.MCU I/O port | 4.LED row pin |
| 5.Power switch | 6.Reset button | 7.Boot mode configuration | 8.SWD interface |

The CH32V203K-R0 EVT board comes with the following resources.

Motherboard - CH32V203K-R0

1. Main control MCU: CH32V203K8T6\ CH32V203K6T6
2. USB interface P6, P7: USB communication interface PA11, PA12 of the main chip
3. MCU I/O port: I/O pinout interface of the main control MCU
4. LED row pin: The row pin connects to the I/O of the main control MCU to control the LED
5. Power switch: Used to cut off or connect external 5V power supply or USB power supply
6. Reset button: For external manual reset power supply switch
7. Boot mode configuration: Select the boot mode when the chip is powered on by configuring BOOT0
8. SWD interface: For downloading, simulation debugging

CH32V203 Evaluation Board



Descriptions

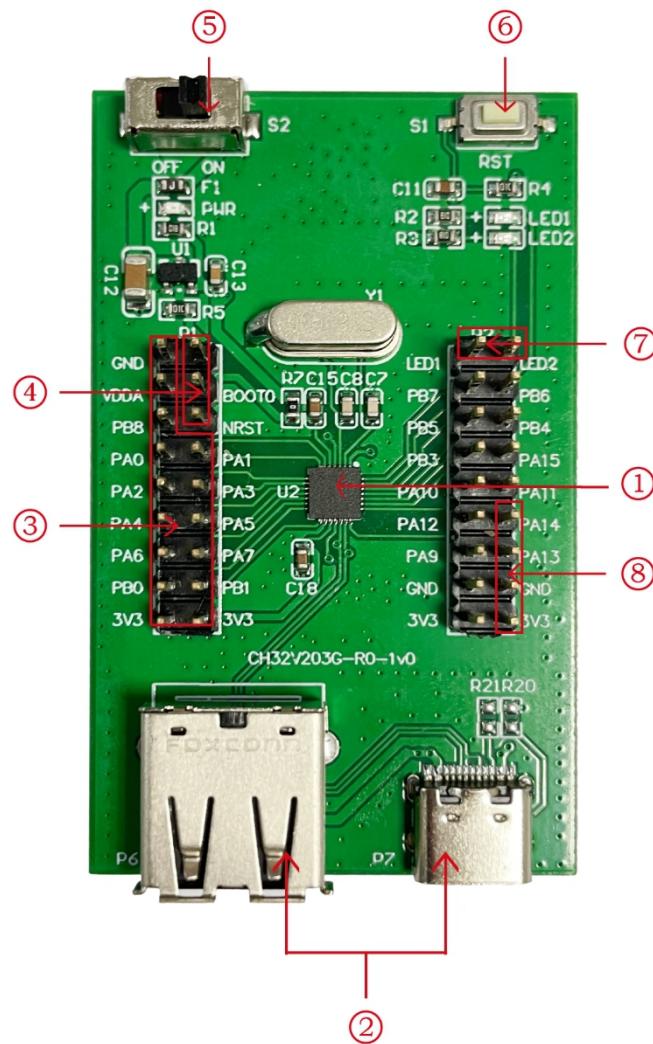
- | | | | |
|--------------------|-----------------|----------------|---------------------------|
| 1.Main control MCU | 2.USB interface | 3.MCU I/O port | 4.Boot mode configuration |
| 5.Power switch | 6.Reset button | 7.LED row pin | 8.SWD interface |

The CH32V203F-R0 EVT board comes with the following resources.

Motherboard - CH32V203F-R0

1. Main control MCU: CH32V203F6P6
2. USB interface P6, P7: USB communication interface PA11, PA12 of the main chip
3. MCU I/O port: I/O pinout interface of the main control MCU
4. Boot mode configuration: Select the boot mode when the chip is powered on by configuring BOOT0
5. Power switch: Used to cut off or connect external 5V power supply or USB power supply
6. Reset button: For external manual reset power supply switch
7. LED row pin: The row pin connects to the I/O of the main control MCU to control the LED
8. SWD interface: For downloading, simulation debugging

CH32V203 Evaluation Board



Descriptions

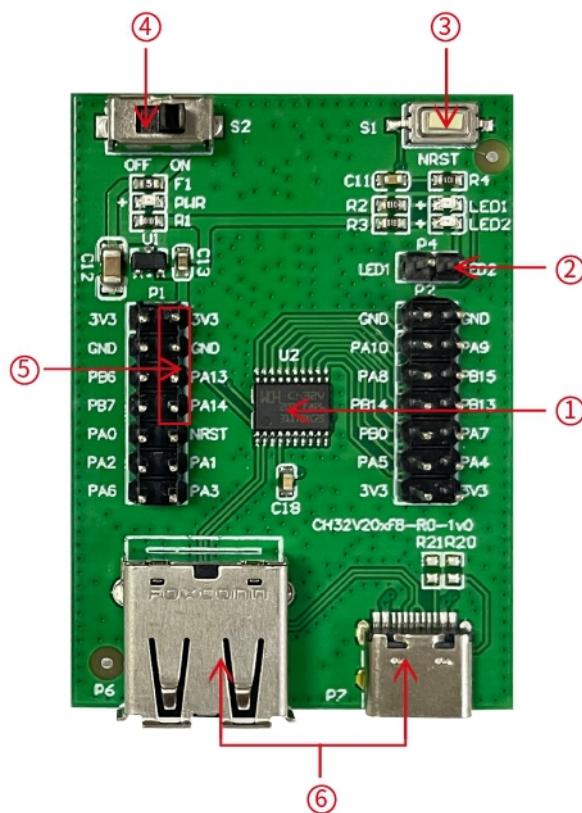
- | | | | |
|--------------------|-----------------|----------------|---------------------------|
| 1.Main control MCU | 2.USB interface | 3.MCU I/O port | 4.Boot mode configuration |
| 5.Power switch | 6.Reset button | 7.LED row pin | 8.SWD interface |

The CH32V203G-R0 EVT board comes with the following resources.

Motherboard - CH32V203G-R0

1. Main control MCU: CH32V203G6U6
2. USB interface P6, P7: USB communication interface PA11, PA12 of the main chip
3. MCU I/O port: I/O pinout interface of the main control MCU
4. Boot mode configuration: Select the boot mode when the chip is powered on by configuring BOOT0
5. Power switch: Used to cut off or connect external 5V power supply or USB power supply
6. Reset button: For external manual reset power supply switch
7. LED row pin: The row pin connects to the I/O of the main control MCU, control the LED
8. SWD interface: For downloading, simulation debugging

CH32V203 Evaluation Board



Descriptions

- | | | |
|---------------------|--------------------|------------------|
| 1. Main control MCU | 2. LED row pin | 3. Reset button |
| 4. Power switch | 5. Debug interface | 6. USB interface |

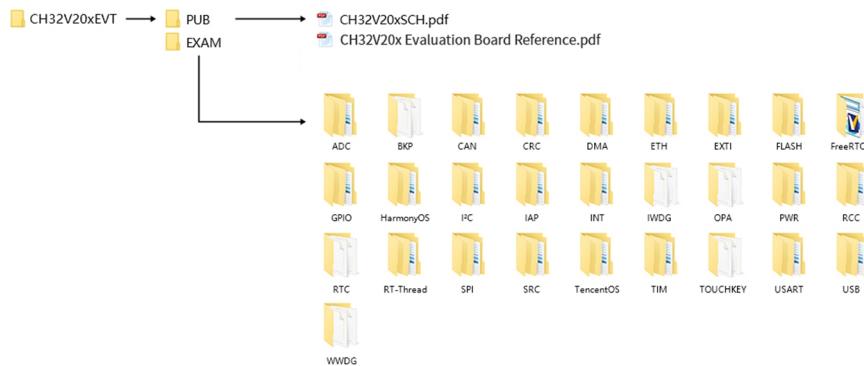
CH32V203F-R0 EVT board comes with the following resources.

Motherboard - CH32V203F-R0

1. Main control MCU: CH32V203F8P6
2. LED row pin: The row pin connects to the I/O of the main control MCU, control the LED
3. Reset button: for external manual reset power supply switch
4. Power switch: Used to cut off or connect external 5V power supply or USB power supply
5. Debug interface: For downloading, simulation debugging
6. USB interface: USB communication interface of the main chip

3. Software Development

3.1 EVT package directory structure



Description:

PUB folder: provides evaluation board manuals, evaluation board schematics.

EXAM folder: Provides software development drivers and corresponding examples for the CH32V20x controller, grouped by peripheral. Each type of peripheral folder contains one or more functional application routines folders.

3.2 IDE use-MounRiver

Download MounRiver_Studio, double click to install it, and you can use it after installation. (MounRiver_Studio instructions are available at the path: MounRiver\MounRiver_Studio\ MounRiver_Help.pdf and MounRiver_ToolbarHelp.pdf)

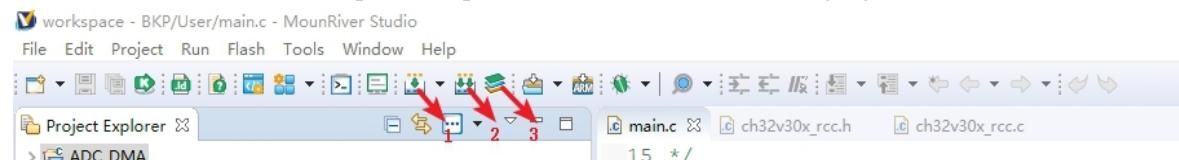
3.2.1 Open project

➤ Open project:

- 1) Double-click project file directly with the suffix name .wvproj under the corresponding project path.
- 2) Click File in MounRiver IDE, click Load Project, select the .project file under the corresponding path, and click Confirm to apply it.

3.2.2 Compilation

MounRiver contains three compilation options, as shown in the following figure.



Compile option 1 is Incremental Build, which compiles the modified parts of the selected project.

Compile option 2 is ReBuild, which performs a global compilation of the selected project.

Compile option 3 is All Build, which performs global compilation for all projects.

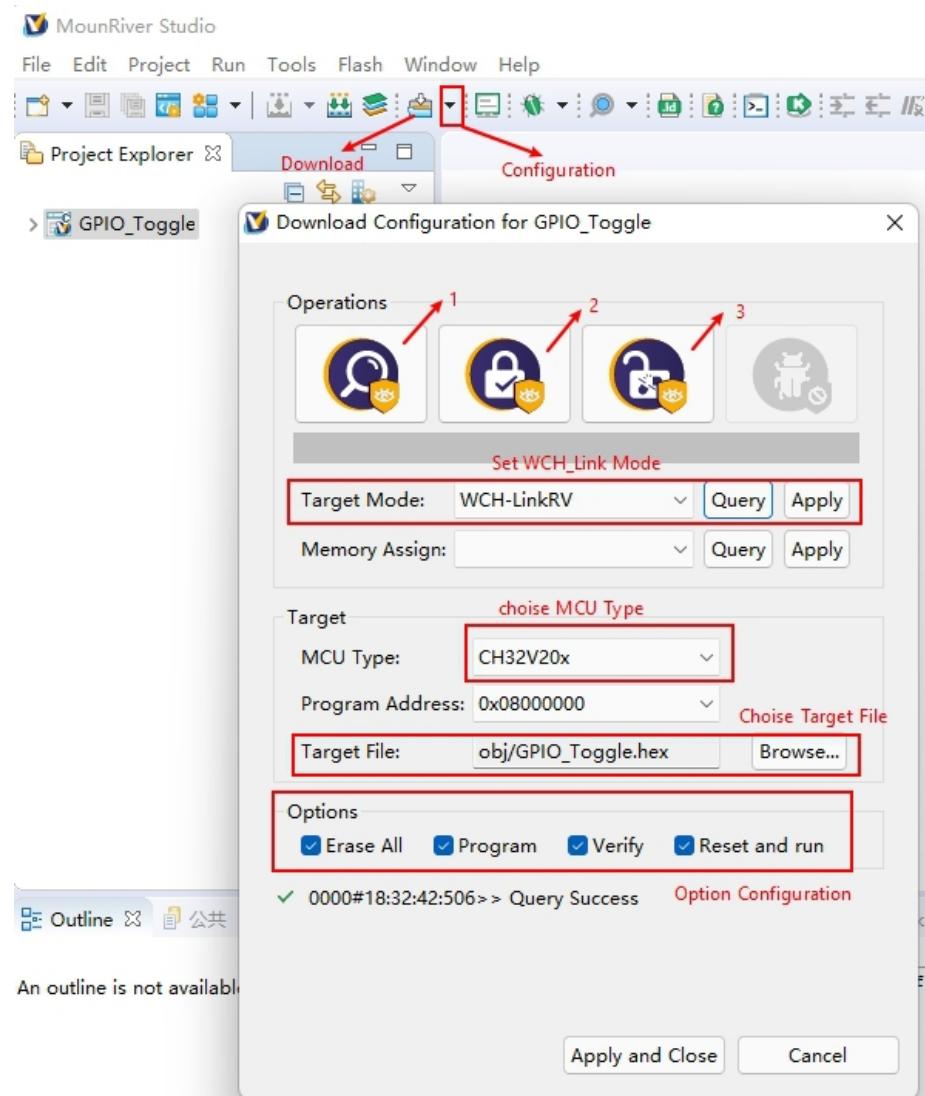
3.2.3 Download/Simulation

➤ Download

- 1) Debugger download

Connect to the hardware via WCH-Link (see WCH-Link instructions for details, path: MounRiver\MounRiver_Studio\ WCH-Link instructions.pdf), click the Download button on the IDE, and select

Download in the pop-up interface, as shown in the figure below.



1 for querying the chip read protection status.

2 for setting the chip read protection and re-powering the configuration to take effect.

3 for lifting the chip read protection and re-powering the configuration to take effect.

➤ Simulation

1) Toolbar description

Click Debug button in the menu bar to enter the download, see the image below, the download toolbar.



Detailed functions are as follows.

- (1) Reset: After reset, the program returns to the very beginning.
- (2) Continue: Click to continue debugging.
- (3) Terminate: Click to exit debugging.
- (4) Single-step jump-in: Each time you tap a key, the program runs one step and encounters a function to enter and execute.
- (5) Single-step skip: jump out of the function and prepare the next statement.
- (6) Single-step return: return the function you jumped into

Instruction set single-step mode: click to enter instruction set debugging (need to use with 4, 5 and 6 functions).

2) Set breakpoints

Double-click on the left side of the code to set a breakpoint, double click again to cancel the breakpoint, set the breakpoint as shown in the following figure;

```

132     * @return  none
133     */
134 int main(void)
135 {
136     u16 i;
137     Set breakpoint
138     Delay_Init();
139     USART_Printf_Init(115200);
140     printf("SystemClk:%d\r\n", SystemCoreClock);
141
142     ADC_Function_Init();
143
144     DMA_Tx_Init(DMA1_Channel1, (u32)&ADC1->RDATA, (u32)TxBuf, 10);
145     DMA_Cmd(DMA1_Channel1, ENABLE);
146
147     ADC_RegularChannelConfig(ADC1, ADC_Channel_2, 1, ADC_SampleTime_241Cycles);
148     ADC_SoftwareStartConvCmd(ADC1, ENABLE);
149     Delay_Ms(50);
150     ADC SoftwareStartConvCmd(ADC1, DISABLE);

```

3) Interface display

(1) Instruction set interface

Click on the instruction set single-step debugging can enter the instruction debugging, to single-step jump in for example, click once to run once, the running cursor will move to view the program running, the instruction set interface is shown as follows.

Address	Instruction	Description
00000540:	auipc a1,0x20000	
00000544:	addi a1,a1,-1344 # 0x20000000 <APBAHBPresc	
00000548:	addi a2, gp, -2024	
0000054c:	bgeu a1, a2, 0x560 <handle_reset+56>	
00000550:	lw t0, 0(a0)	
00000554:	sw t0, 0(a1)	
00000558:	addi a0, a0, 4	Running cursor
0000055a:	addi a1, a1, 4	
0000055c:	bltu a1, a2, 0x550 <handle_reset+40>	
00000560:	addi a0, gp, -2024	
00000564:	addi a1, gp, -2004	
00000568:	bgeu a0, a1, 0x576 <handle_reset+78>	
0000056c:	sw zero, 0(a0)	
00000570:	addi a0, a0, 4	
00000572:	bltu a0, a1, 0x56c <handle_reset+68>	
00000576:	li t0, 31	
00000578:	csrw 0xbco, t0	
0000057c:	li t0, 11	
0000057e:	csrw 0x804, t0	
00000582:	lui t0, 0x6	
00000586:	addi t0, t0, 136 # 0x6088	
0000058a:	csrs mstatus, t0	
0000058e:	auipc t0, 0x0	

(2) Program running interface

It can be used with instruction set single-step debugging, still take single-step jumping in as an example, click once to run once, the running cursor will move to view the program running, the program running

interface is shown as follows.

```

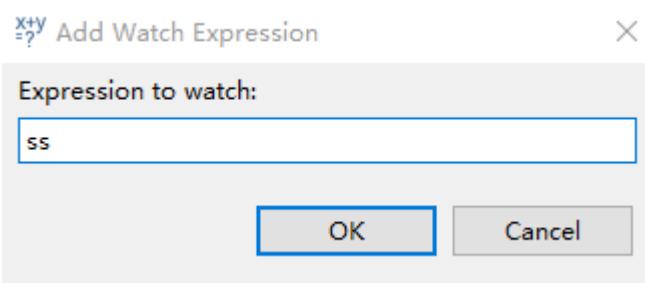
46    "
47    * @return  none
48    */
49 int main(void)
50 {
51     u8 i = 0;
52
53     NVIC_PriorityGroupConfig(NVIC_PriorityGroup_2);
54     Delay_Init();
55     USART_Printf_Init(115200);
56     printf("SystemClock:%d\r\n", SystemCoreClock);
57
58     printf("GPIO Toggle TEST\r\n");
59     GPIO_Toggle_INIT();
60
61     while(1)
62     {
63         Delay_Ms(250);
64         GPIO_WriteBit(GPIOD, GPIO_Pin_0, (i == 0) ? (i = Bit_SET) : (i = Bit_RESET));
65     }
66 }
67

```

Running cursor

4) Variables

Hover over the variable in the source code to display the details, or select the variable and right-click add watch expression



Fill in the variable name, or just click OK to add the variable you just selected to the pop-up.

Expressions		
Expression	Type	Value
(x)= ss	u16	10
+ Add new expres.		

5) Peripheral registers

In the lower left corner of IDE interface Peripherals interface shows a list of peripherals, tick the peripherals will display its specific register name, address, value in the Memory window.

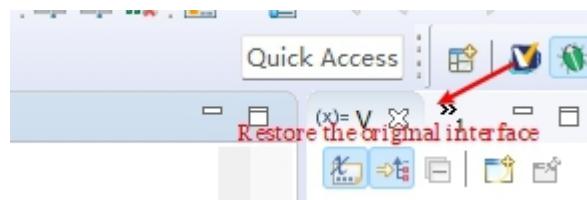
The screenshot shows the software interface for the CH32V20x Evaluation Board. At the top, there are three tabs: Breakpoints, Modules, and Peripherals. The Peripherals tab is selected, displaying a list of peripherals with their addresses and descriptions. Below this is the Registers tab, which is also selected, showing the register structure for the ADC_TKEY peripheral.

Peripheral	Address	Description
ADC_TKEY	0x40012400	Analog to digital converter
AFIO	0x40010000	Alternate function I/O
BKP	0x40006C00	Backup registers
CRC	0x40023000	CRC calculation unit
DBG	0xE0042000	Debug support
DMA	0x40020000	DMA controller
EXTEND	0x40023800	Extend configuration
EXTI	0x40010400	EXTI
FLASH	0x40022000	FLASH
GPIOA	0x40010800	General purpose I/O
GPIOB	0x40010C00	General purpose I/O
GPIOC	0x40011000	General purpose I/O
GPIOD	0x40011400	General purpose I/O
I2C1	0x40005400	Inter integrated circuit

Register	Address
ADC_TKEY	0x40012400
STATR	0x40012400
CTLR1_CTLR	0x40012404
CTLR2	0x40012408
SAMPTR1	0x4001240C
SAMPTR2	0x40012410
IOFR1	0x40012414
IOFR2	0x40012418
IOFR3	0x4001241C
IOFR4	0x40012420
WDHTR	0x40012424
WDLTR	0x40012428
RSQR1	0x4001242C
RSQR2	0x40012430
RSQR3_CHANNFI	0x40012434

Note:

(1) When debugging, click the icon in the upper right corner to enter the original interface.



(2) For documentation to access the compiler, click F1 to access the help documentation for detailed instructions.

➤ Engineering Chip Selection

In the project there are a variety of chip selection, the chip used in the CH32V20x development board CH32V203C8T6 as an example of engineering chip selection compiled to achieve different peripheral functions, the steps are as follows.

- 1) Compile the project chip selection according to different priorities. Click on the ch32v20x.h file in the Peripheral—> inc folder to check the chip type, as shown below, to see the different chip project definitions.

```

ch32v20x_rcc.h
ch32v20x_rtc.h
ch32v20x_gpi.h
ch32v20x_sm.h
ch32v20x_usart.h
ch32v20x_wwdg.h
ch32v20x_k.h

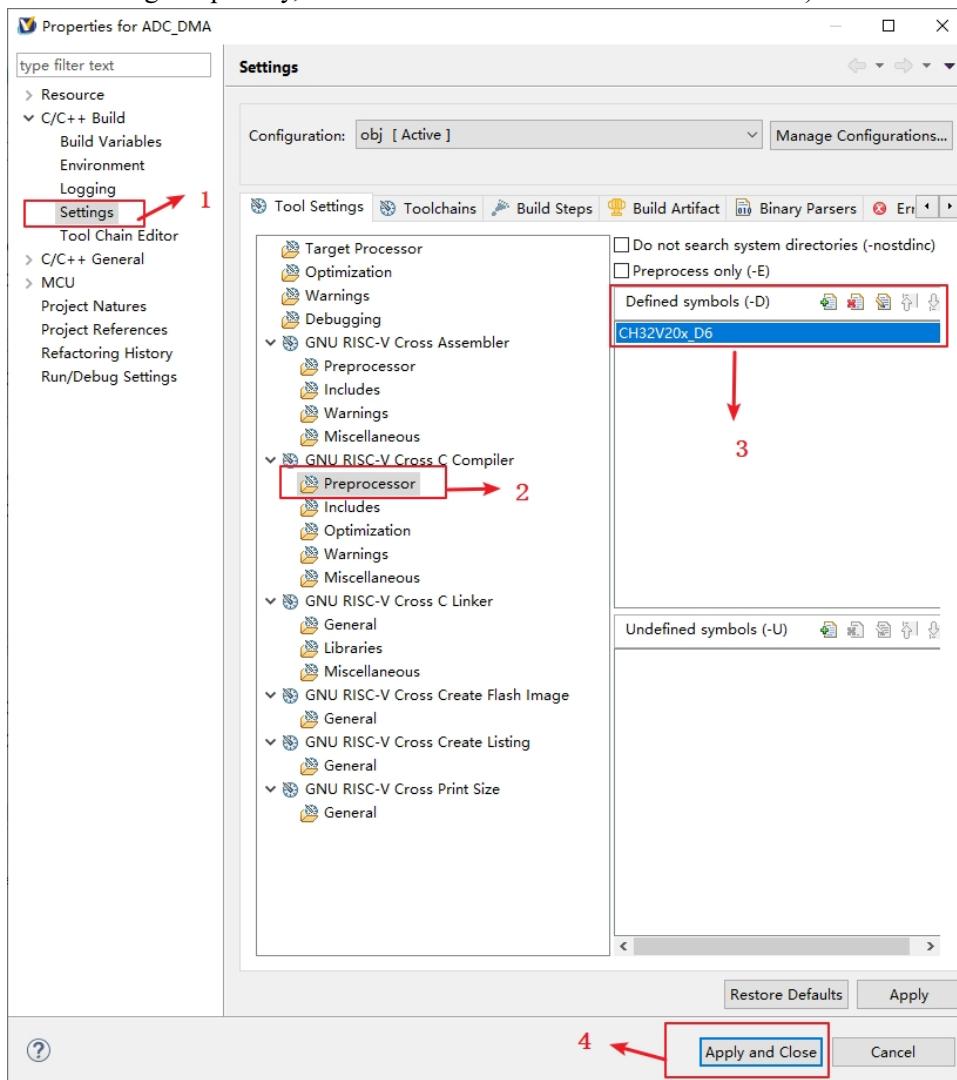
16
17 #if !defined(CH32V20x_D8W) && !defined(CH32V20x_D8) && !defined(CH32V20x_D8)
18 #define CH32V20x_D6           /* CH32V203F6-CH32V203F8-CH32V203G6-CH32V203G8-CH32V203K6-CH32V203K
19 // #define CH32V20x_D8          /* CH32V203RBT6 */
20 // #define CH32V20x_D8W         /* CH32V208 */
...

```

Method 1: Click Project Properties, as shown in the following figure, enter the project definition project properties, and click OK to apply.

Method 2: As shown above, check the box according to the defined chip type, for example, if you choose CH32V203C8T6 chip, select CH32V20x_D6.

(Note: Method 1 is the highest priority, if method 1 is used then method 2 is invalid.)



- 2) Select the startup file, click on the Startup folder, and select the startup file corresponding to the definition, as shown in the figure below. Since step 1 selected CH32V20x_D6, the startup_ch32v20x_D6.S file is selected (Note: Different startup files are selected according to the different statements in step 1.)



- 3) Change the starting file memory, click LD file, as shown in the following figure, according to the chip type to choose a different memory configuration, such as the use of chip type CH32V203C8T6, so choose CH32V20x_D6 FLASH and RAM configuration.

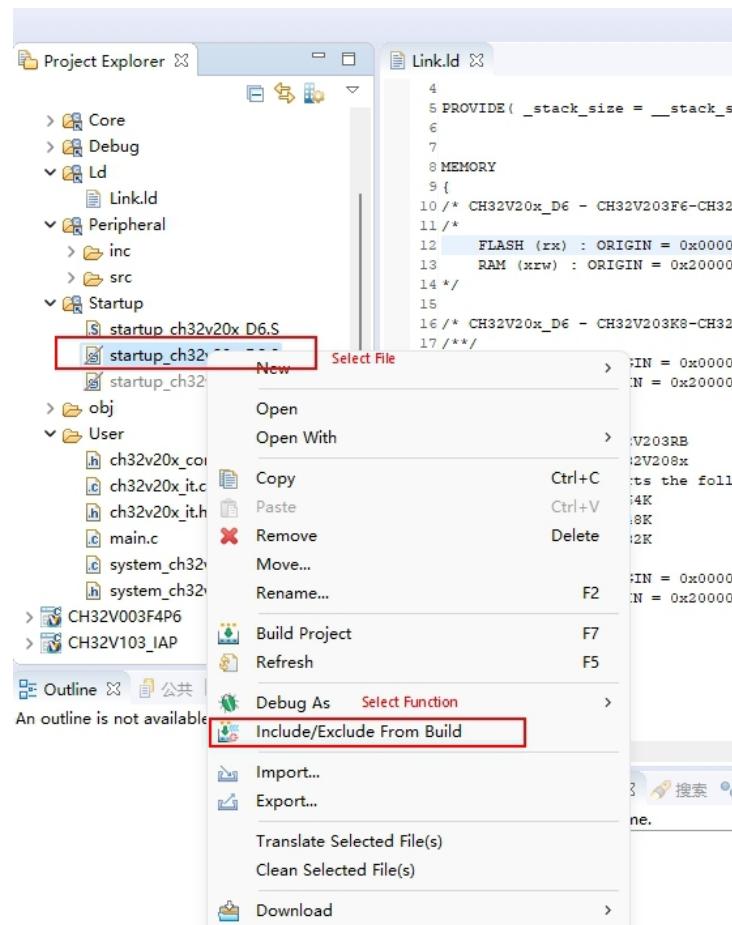
The screenshot shows the WCH-IC Project Manager interface. On the left is the Project Explorer, which contains several project components like ADC_DMA, Debug, Ld, Peripheral, Startup, obj, and User. The Startup folder is expanded, showing files such as startup_ch32v20x_D6.S, startup_ch32v20x_D8.S, startup_ch32v20x_D8W.S, and others. The right side of the interface is the Link.ld editor. It displays assembly code for memory configuration, specifically for the CH32V203C8 chip. A red box highlights a section of the code related to memory regions and configurations. Below the code, a note says "CH32V203C8 Configuration".

```

4
5 PROVIDE( _stack_size = __stack_size );
6
7
8 MEMORY
9 {
10 /* CH32V20x_D6 - CH32V203F6-CH32V203G6-CH32V203K6-CH32V203C6 */
11 /*
12   FLASH (rx) : ORIGIN = 0x00000000, LENGTH = 32K
13   RAM (xrw) : ORIGIN = 0x20000000, LENGTH = 10K
14 */
15
16 /* CH32V20x_D6 - CH32V203K8-CH32V203C8-CH32V203G8-CH32V203F8 */
17 /**
18   FLASH (rx) : ORIGIN = 0x00000000, LENGTH = 64K
19   RAM (xrw) : ORIGIN = 0x20000000, LENGTH = 20K
20
21
22 /* CH32V20x_D8 - CH32V203RB
23 CH32V20x_D8W - CH32V208x
24 FLASH + RAM supports the following configuration
25 FLASH-128K + RAM-64K
26 FLASH-144K + RAM-48K
27 FLASH-160K + RAM-32K
28
29   FLASH (rx) : ORIGIN = 0x00000000, LENGTH = 160K
30   RAM (xrw) : ORIGIN = 0x20000000, LENGTH = 32K
31 */
32 }
33
34
35 SECTIONS
36 {
37

```

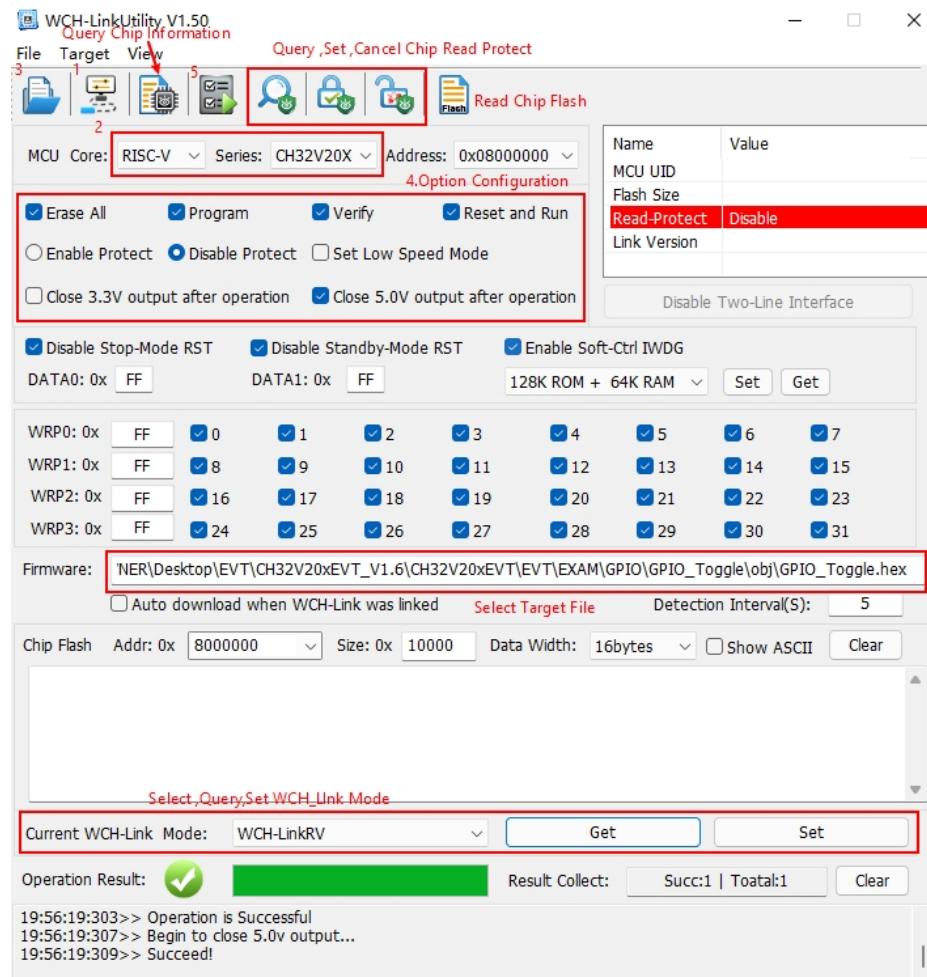
Note: For conflicting/useless files, it is recommended to use the Engage/Exclude Compile function to give a brief description of the files in the Startup folder. First, select the project file in the folder, right-click, and select the Participate/Exclude compile function button, take the startup_ch32v20x_D8.S file in the figure as an example, if the project status is Exclude compile, click this function to Participate compile. Similarly, if the project status is involved in the compilation, click this function to exclude the compilation. (Note: Folder can also be applied).



4. WCH-LinkUtility.exe Download

The download process for the chip using the WCH-LinkUtility tool is:

- 1) Connect WCH-Link
- 2) Select chip information
- 3) Add firmware
- 4) If the chip is read protected, you need to release the chip read protection.
- 5) Execute

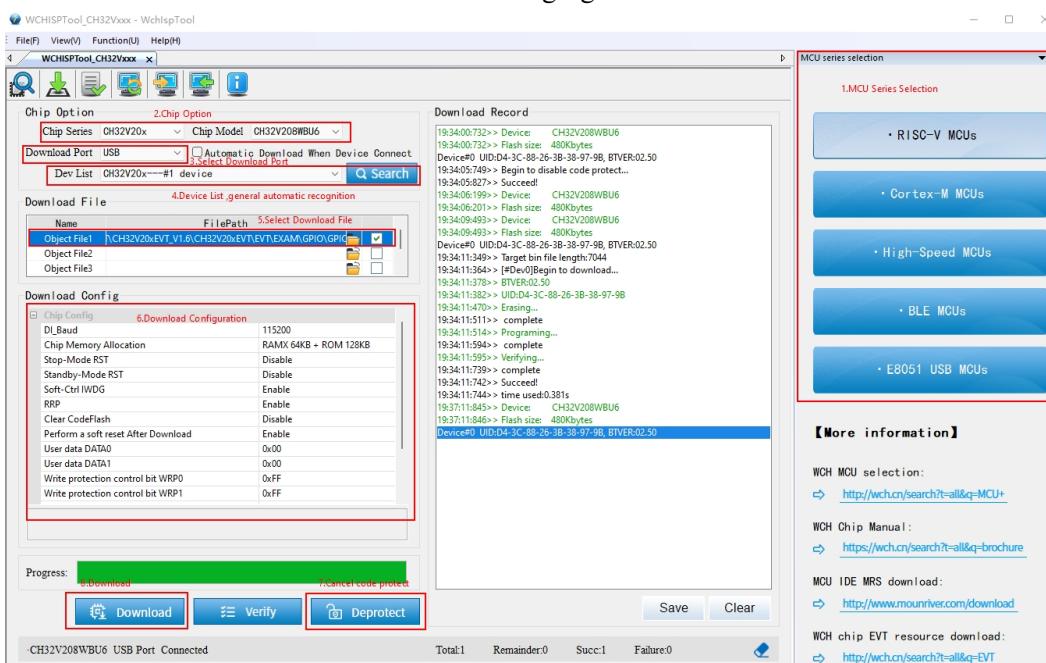


5. WCHISPTool.exe Download

The WCHISPTool tool is used to download the chip, supporting both USB and serial port. the USB pins are PA11 (DM), PA12 (DP) or PB6 (DM), PB7 (DP), and the serial port pins are PA9 (TX), PA10 (RX). The download process is:

- (1) BOOT0 is connected to VCC, BOOT1 is connected to ground, and the PC is connected via serial port or USB.
- (2) Open the WCHISPTool tool, select the appropriate download method, choose to download the firmware, check the chip configuration and click download.
- (3) Ground BOOT0, reapply power and run APP program.

The WCHISPTool tool interface is shown in the following figure.



1. Select MCU series and chip model
2. Select the serial port download mode
3. Identify the device, usually automatically, if it fails to identify, you need to select manually
4. Select the firmware, select the downloaded .hex or .bin target program file
5. Configure the download according to the requirements
6. Click download

6. Statement of attention

WCH official website: <https://wch-ic.com/>

WCH-LINK instructions for use: <https://www.wch-ic.com/products/WCH-Link.html>