

HT32init 使用手册

版本：V1.09 日期：2020-05-06

www.holtek.com

目录

1. 简介与安装.....	3
1.1 简介.....	3
1.2 安装.....	3
2. 快速上手.....	3
3. “Clock”页面的设置.....	7
4. “AFIO”页面的设置.....	8
5. “GPIO”页面的设置.....	9
6. “EXTI”页面的设置.....	10
7. “TM”页面的设置.....	11
8. “SCTM”页面的设置.....	13
9. “ADC/DAC”页面的设置.....	14
9.1 ADC 的说明：.....	14
9.2 DAC 的说明：.....	15
10. “CMP”页面的设置.....	16
11. “SPI”页面的设置.....	17
12. “UART/USART”页面的设置.....	18
13. “I2C”页面的设置.....	19
14. “SCI”页面的设置.....	20
15. “EBI”页面的设置.....	21
15.1 EBI 基础设置：.....	21
15.2 驱动 TFT 屏：.....	22
15.3 LCD 平台：.....	22
16. “I2S”页面的设置.....	26
17. “SDIO”页面的设置.....	27
18. “CSIF”页面的设置.....	27
19. “MonoLCD”页面的设置.....	28
20. “Other”页面的设置.....	29
20.1 CKOUT 设置：.....	29
20.2 PDMA 设置：.....	29
20.3 UART IAP 设置：.....	30
20.4 USB IAP 设置：.....	32
20.4.1 USB HID IAP.....	32
20.4.2 USB mass storage IAP.....	36
21. 版本历史.....	39

1. 简介与安装

1.1 简介

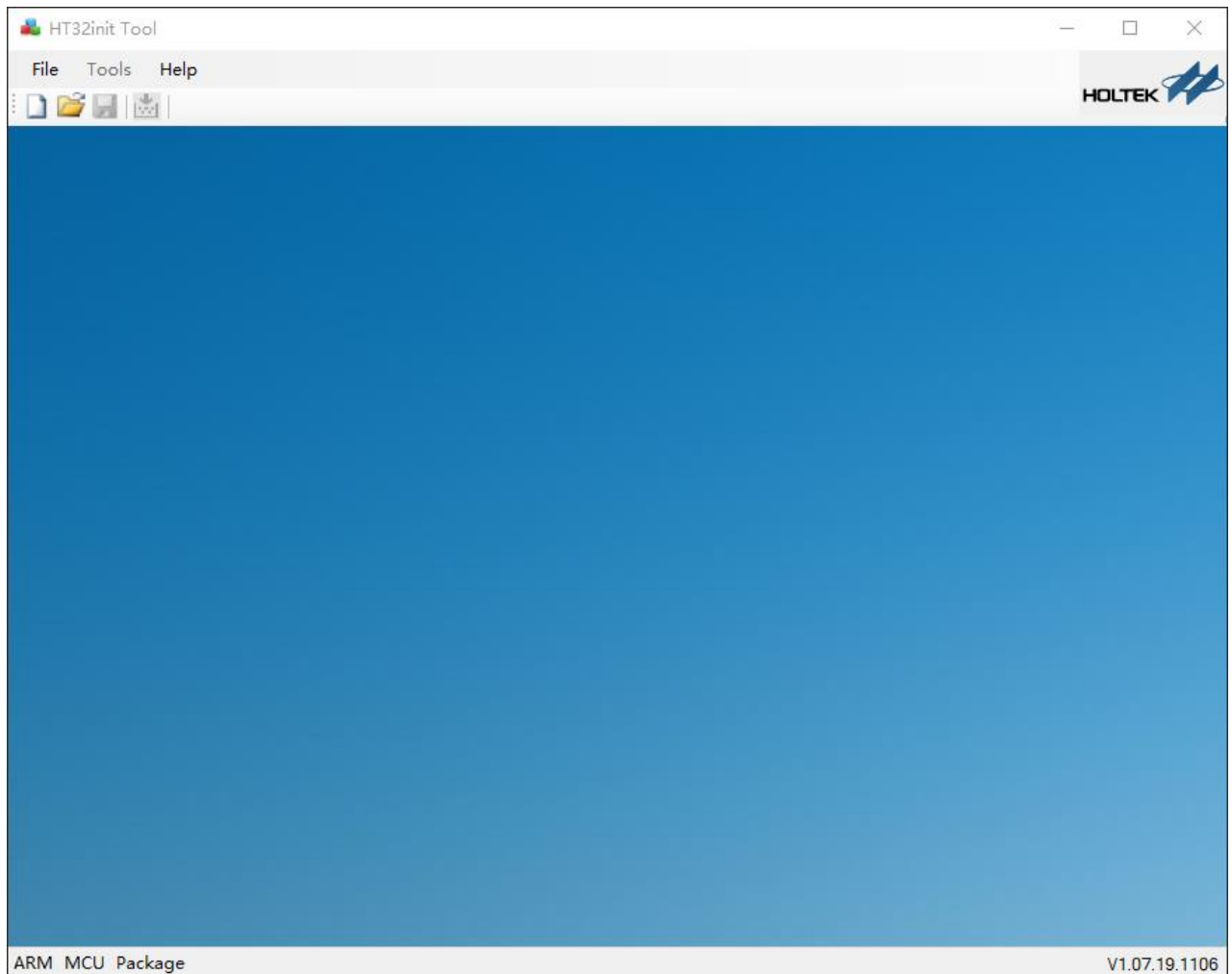
HT32init 是一个 HT32 系列 MCU 的辅助开发软件，可以协助建立不同型号 MCU 的 MDK 工程，并可以按图形化设定选项，完成 MCU 基础外设的初始化。除此之外还附加了 USB/UART IAP 功能，LCD 开发平台，FatFs 等功能。

1.2 安装

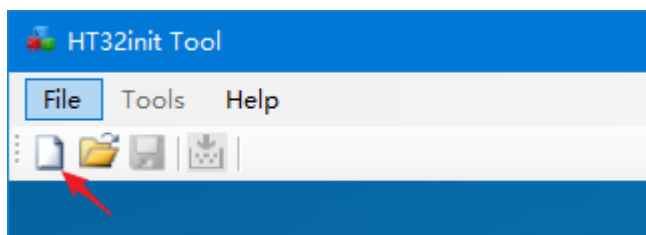
- 安装于 Microsoft Win7/WIN8/WIN10 桌面操作系统.
- 系统需要安装 .NET Framework 4 或者更新版本。一般 WIN10 都已经随系统安装，若没有安装，请于 Microsoft 官网下载安装：<https://dotnet.microsoft.com/download>
- 具体安装过程里，按提示一直点“下一步”就可以完成安装.

2. 快速上手

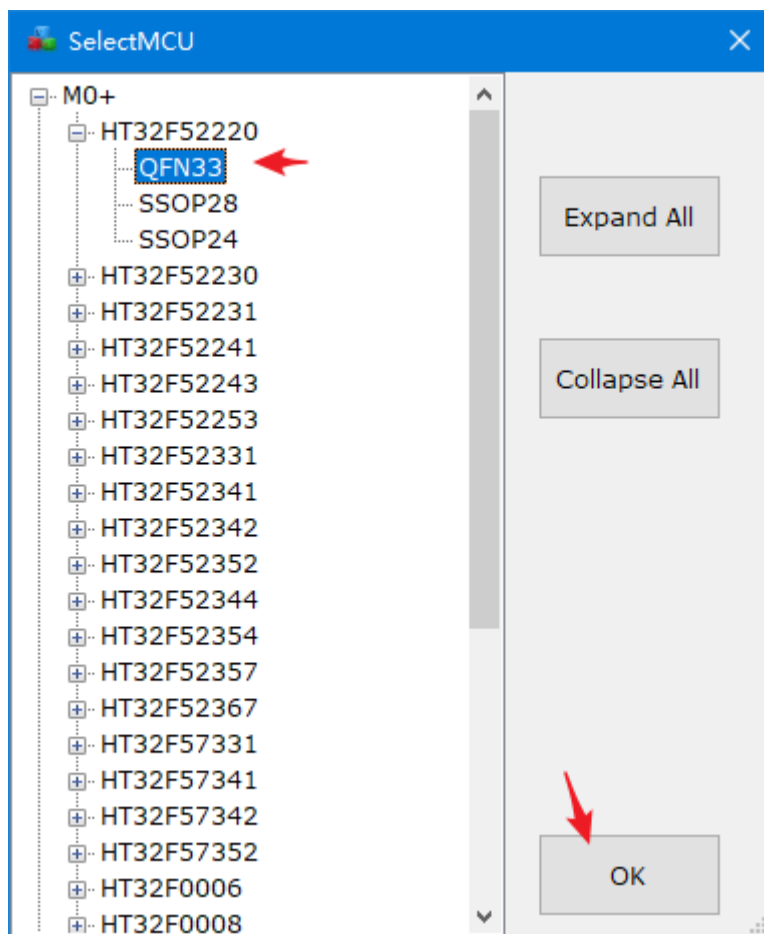
2.1 打开安装好的 HT32Init.exe，初始界面如下图



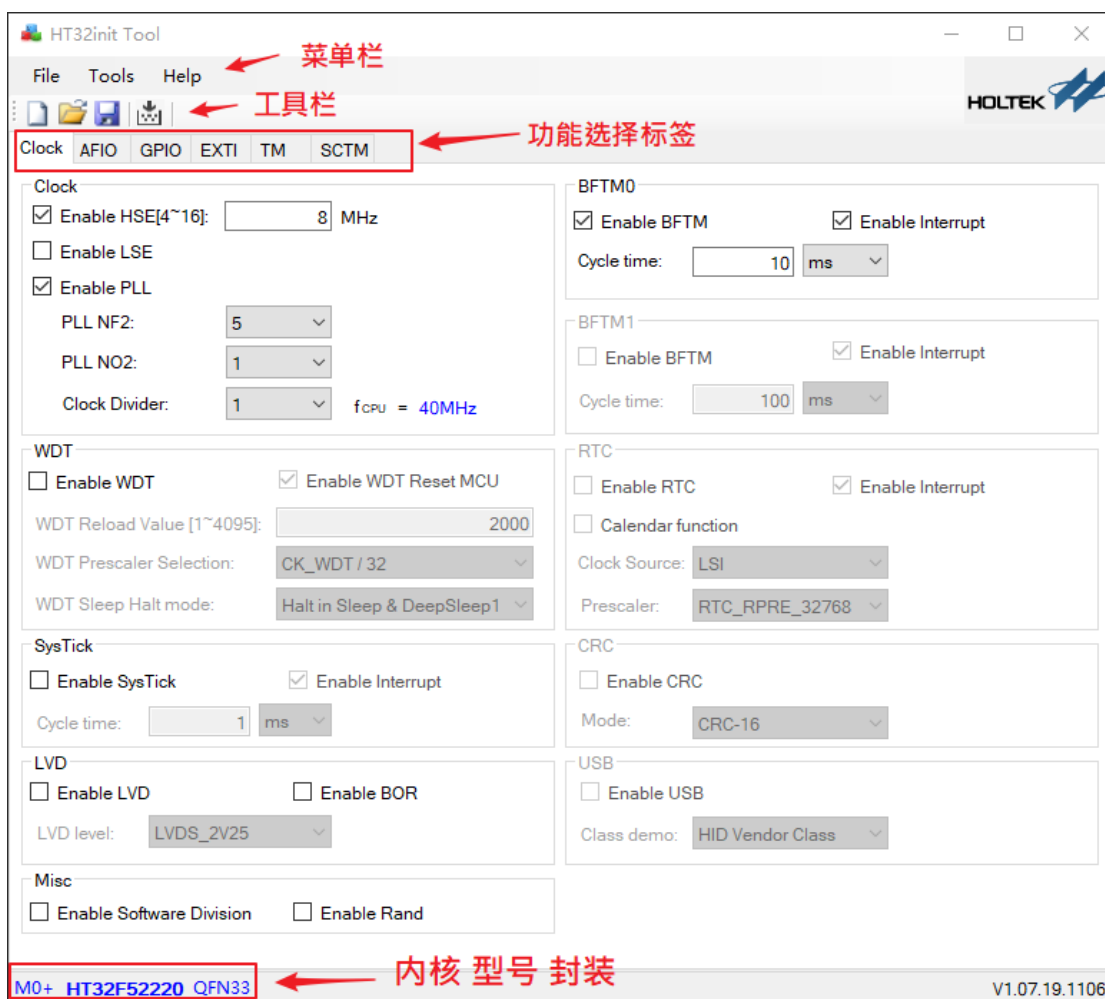
2.2 选择菜单 File→New, 或者点击图标 New, 或者鼠标点击主窗口任意位置[窗口蓝色部分]



2.3 在弹出的型号/封装选择窗口里, 选择所需的型号和封装, 例如下图的 HT32F52220, QFN33, 点 OK



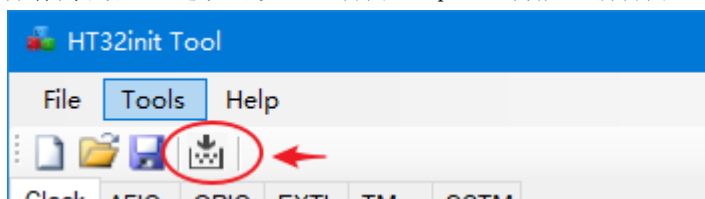
2.4 主窗口会变成下图所示



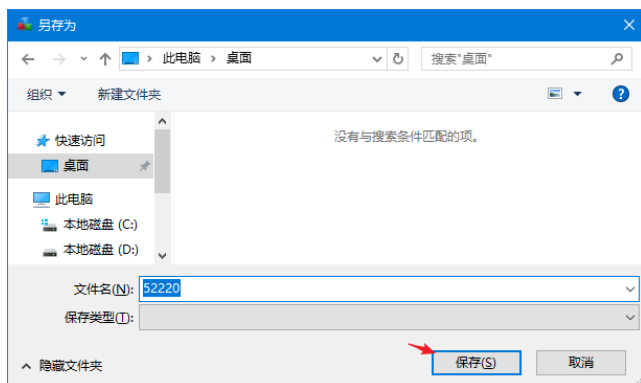
2.5 选择菜单 Tools→Build, 或者点击图标 Build。

保存 (Save) 功能是用以保存所有配置选项，并不输出 MDK 工程。

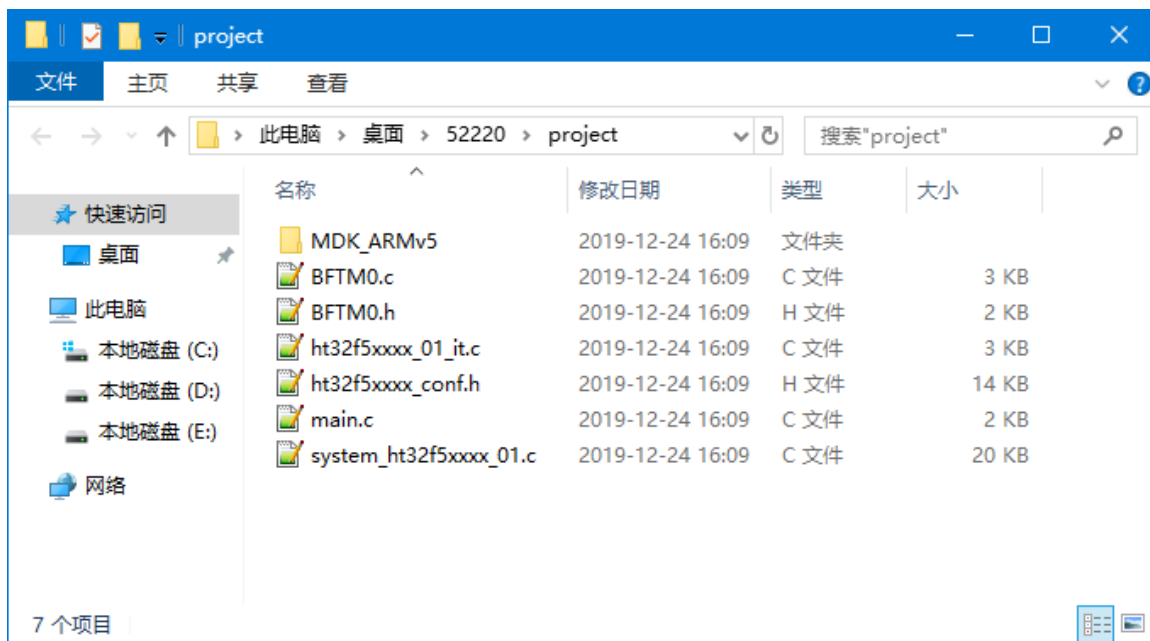
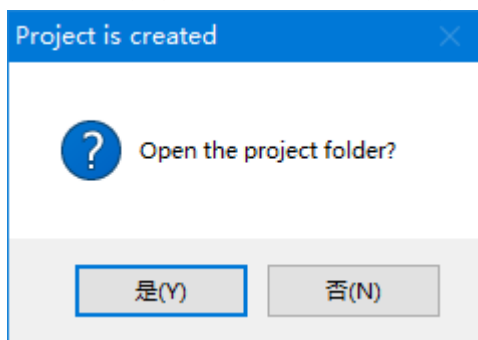
保存好的配置选项可以通过打开 (Open) 功能重新打开。



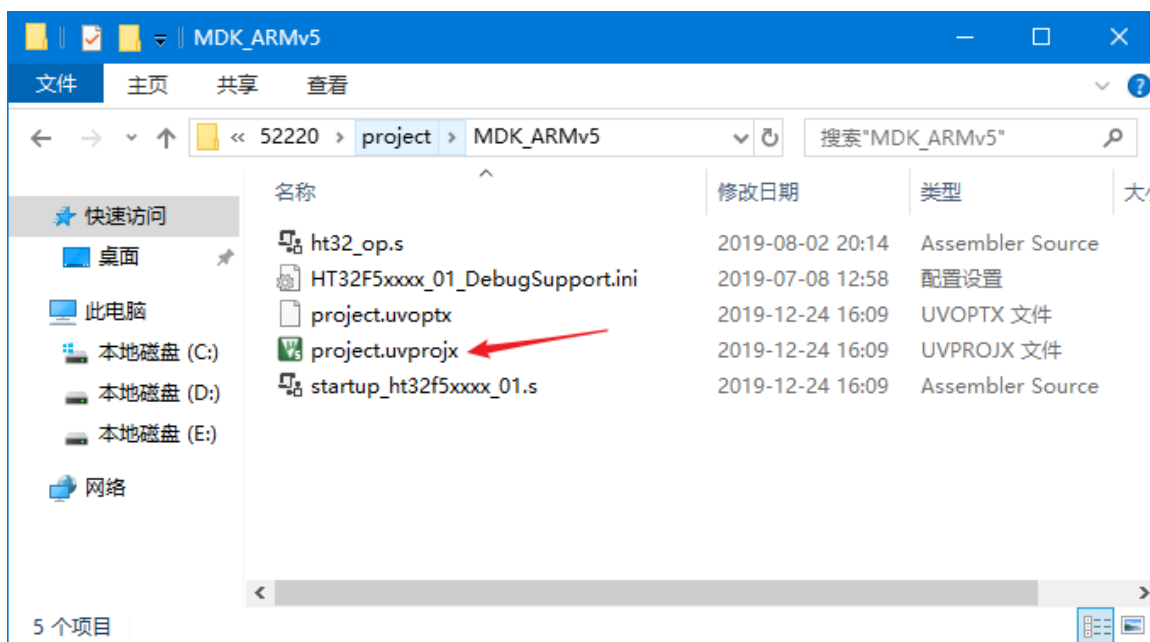
2.6 在弹出的保存窗口，选择保存路径，输入名称，保存



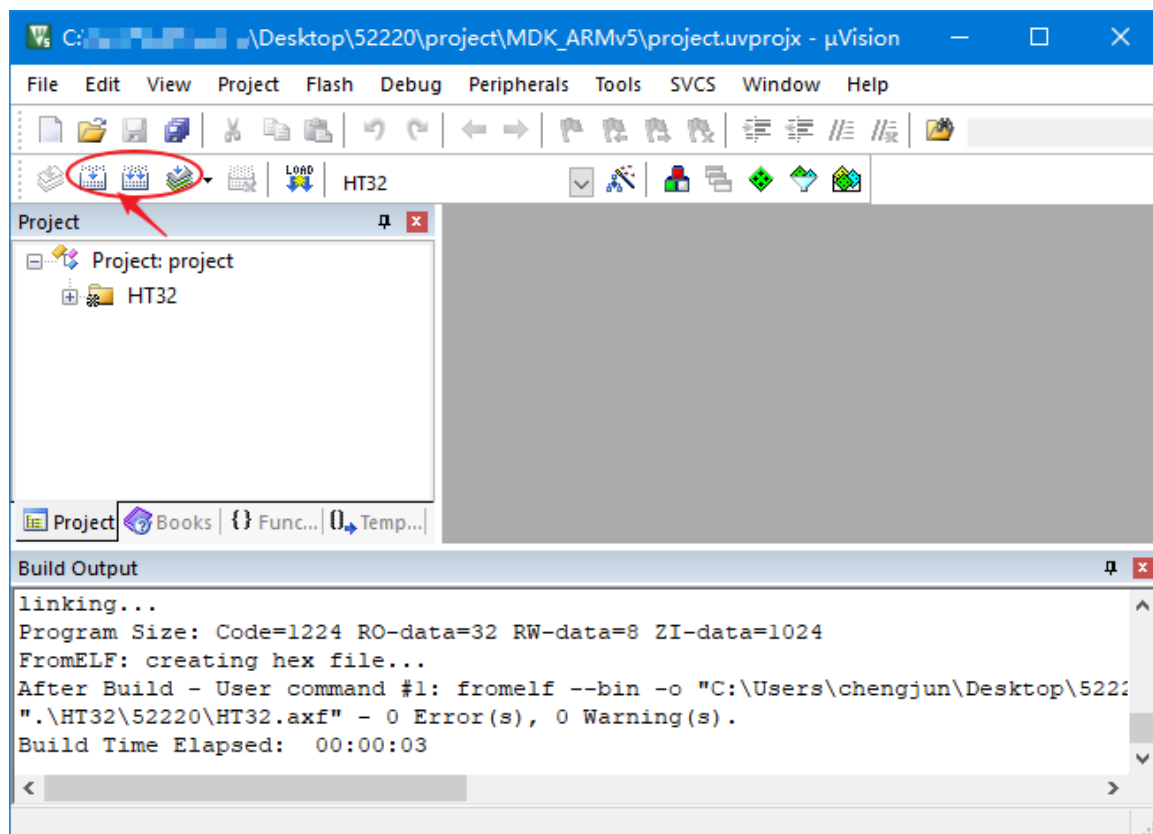
2.7 工程创建好会弹出提示打开对应文件夹的对话框，按提示打开



2.8 进入 MDK_ARMv5 文件夹，如果已经安装了 MDK, 双击 project.uprojx 就可以启动 MDK 打开工程了

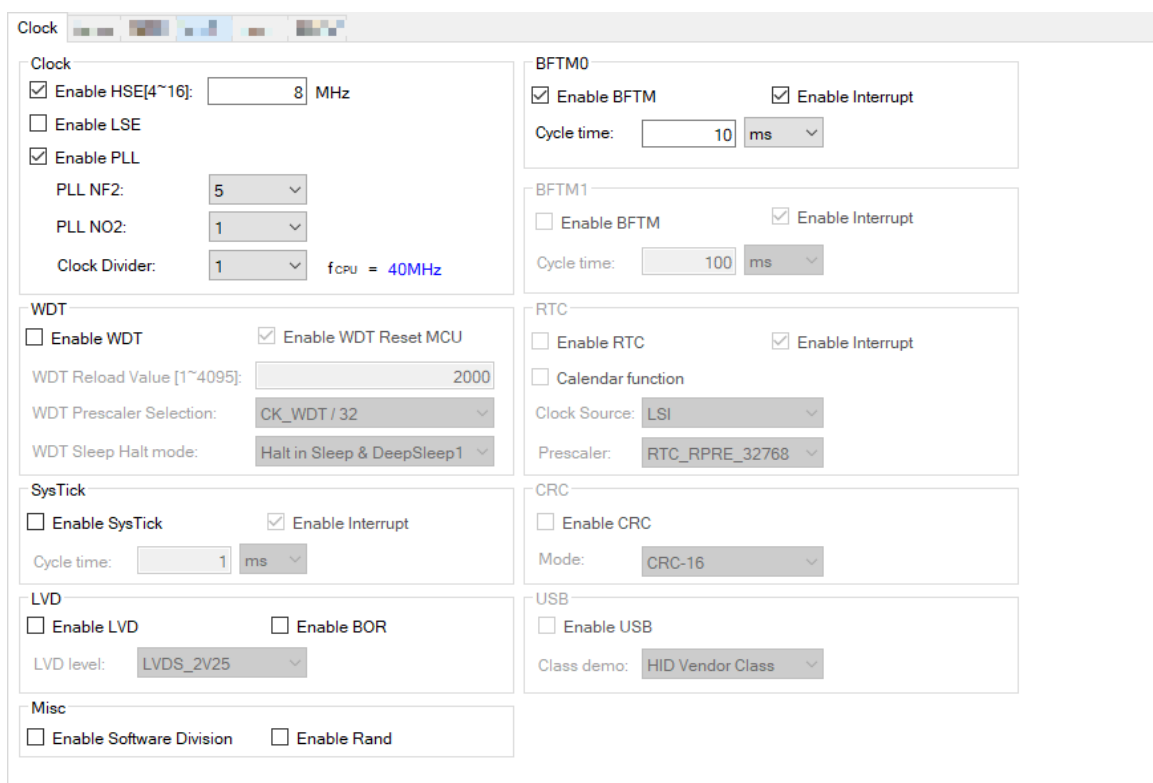


2.9 点击 MDK 的 build 工具编译工程。编译完成后，若对应硬件已经通过 elink32 lite/pro 连接，就能点击 download 下载到 MCU 内了。



3. “Clock”页面的设置

“Clock”页面不只有晶振/PLL/分频器的设置，还包括有WDT, SysTick, LVD, BFTM, RTC, CRC, USB 和 Misc 的设置，如下图。



- #### 4. “AFIO”页面的设置

AFIO 页面是最主要的设置页面。AFIO 有 0~15 一共 16 种功能选择, 包含有 default, GPIO, ADC, CMP, GPTM/MCTM, SPI, UART/USART, I2C, SCI, EBI, I2S, SDIO, CSIF, MonoLCD, Other system。不是每个型号的 MCU 都包含以上所有功能, 请对照 datasheet 的引脚分配章节, 其中的分配表与 AFIO 页面是一致的。

[illegible]

当需要某个功能，请用鼠标双击对应位置，例如双击 I2C_SCL/I2C_SDA,双击的位置将会变成绿色，功能标签也会多显示出 I2C，详见后面章节的“I2C 页面的设置”。

5. “GPIO”页面的设置

GPIO 页面用于设置输入输出方向，输入是否使能，上拉/下拉选择，输出状态，开路输出和输出电流。

Clock	AFIO	GPIO						
PIN NO.	GPIO Name	Direction	Input Enable	Pull State	Output State	Open Drain	Current	
1	PA0	Input	Disable	NoPull	Low	Disable	4mA	
2	PA1	Input	Enable	PullUp	Low	Disable	4mA	
3	PA2	Output	Disable	NoPull	High	Enable	16mA	
4	PA3	Input	Disable	NoPull	Low	Disable	4mA	
5	PA4	Input	Disable	NoPull	Low	Disable	4mA	
6	PA5	Input	Disable	NoPull	Low	Disable	4mA	
7	PA6	Input	Disable	NoPull	Low	Disable	4mA	
8	PA7	Input	Disable	NoPull	Low	Disable	4mA	
9	CLDO							
10	VDD							
11	VSS							
12	nRST							
13	NC							
14	NC							
15	PB12	Input	Disable	NoPull	Low	Disable	4mA	
16	PB13	Input	Disable	NoPull	Low	Disable	4mA	
17	PB14	Input	Disable	NoPull	Low	Disable	4mA	
18	PA9	Input	Disable	NoPull	Low	Disable	4mA	
19	PA12	Input	Disable	NoPull	Low	Disable	4mA	
20	PA13	Input	Disable	NoPull	Low	Disable	4mA	
21	PA14	Input	Disable	NoPull	Low	Disable	4mA	
22	PA15	Input	Disable	NoPull	Low	Disable	4mA	

- 绿色表示在 AFIO 页面里，对应的 IO 已经被设置成 GPIO 功能；
- 灰色表示在 AFIO 页面里，对应的 IO 没有被设置成 GPIO 功能，或者是不可设置的引脚，例如 VDD, VSS 就不能设置。
- 当设置输入 IO 时，应该设置的列有 Direction, Input Enable, Pull State。
- 当设置输出 IO 时，应该设置的列有 Direction, Output State, Open Drain 和 Current。
- 修改过的位置将会标示为黄色。

6. “EXTI”页面的设置

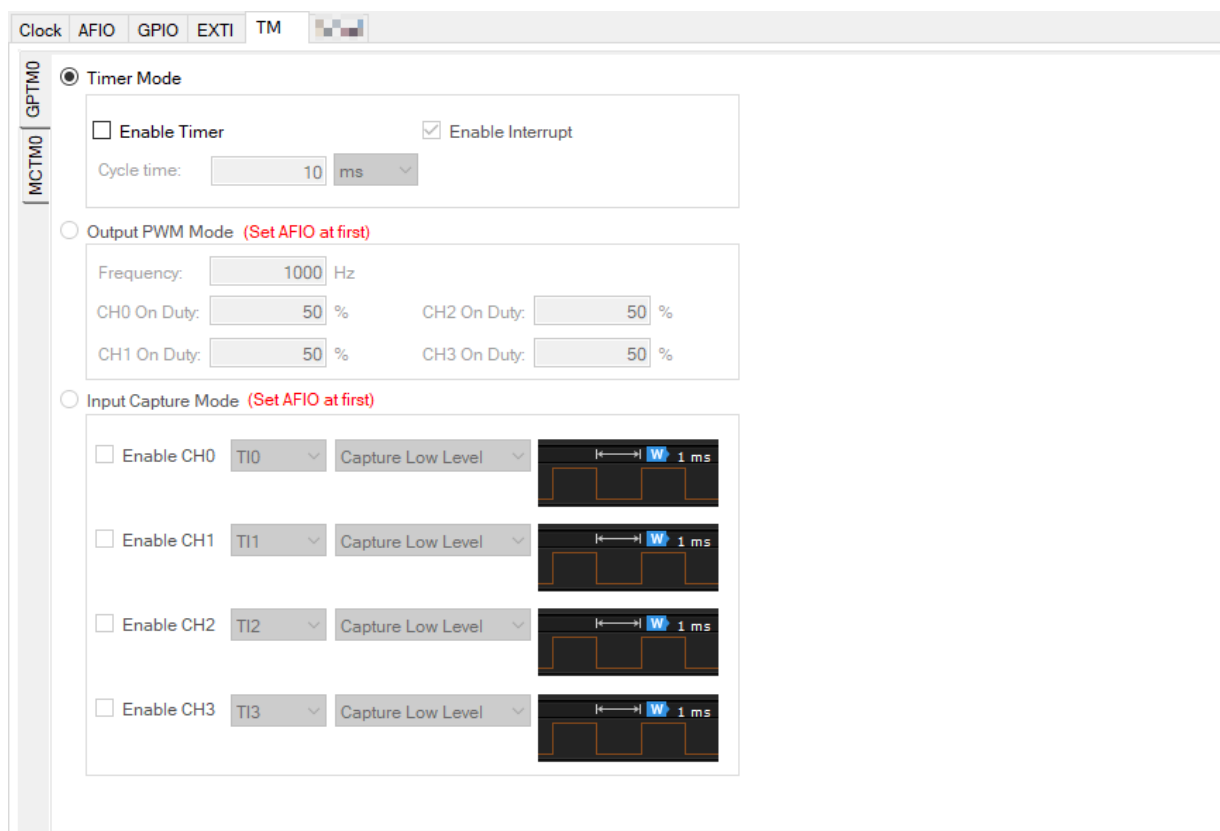
EXTI 有从 0~15 一共 16 个 channel，可以选择映射到哪一个脚，可以选择是用于触发中断还是触发唤醒功能，中断的类型，debounce 以及唤醒电平类型。

Clock	AFIO	GPIO	EXTI			

- 当一个 IO 在 AFIO 和 GPIO 页面里设置成 GPIO + input 模式, 就可以在本页面的 Port 列里刷新出来, 例如 PA2 被设置成 output 了, 那 EXTI2 的 port 就没有 PA2.
- EXTI0 只能选择 PA0, PB0, PC0, 或者是 PD0, 其中某一个。同理, EXTI1 只能选择 PA1, PB1, PC1, 或者是 PD1, 其中某一个。
- Purpose 可以选择用于产生外部输入中断, 唤醒, 或者是两种都有。
- 中断类型可以设置为高电平, 低电平, 上升沿, 下降沿, 或者是两种边沿都触发。
- Debounce 是设置中断的抖动抑制, 可以在 0~65535 之间设置, 越大则抑制抖动的能力越强, 不过也会带来迟滞, 需要合理设置。
- 唤醒类型可以选择高电平, 或者低电平。

7. “TM”页面的设置

TM 页面左侧会显示出选用的 MCU 所有 GPTM,MCTM,PWM。52220 就只会显示 GPTM0，因为它没有 MCTM 和 PWM。



- 如果 AFIO 页面里未选择 GPTM, 则本页面里 GPTM 只有基础的 Timer Mode, 用于基础的定时器功能, 可设置是否使能, 是否产生中断, 以及中断周期。
- 如果 AFIO 页面里选择了 GT_CH2, 则本页面可以设置为 PWM 输出模式或者输入捕捉模式。

Clock	AFIO	GPIO	EXTI	TM	SCTM	
	PIN NO.	AF0 Default	AF1 GPIO	AF2 ADC	AF4 TM	AF5 SPI
	1	PA0		ADC_IN2	GT_CH0	SPI1_SCK
	2	PA1		ADC_IN3	GT_CH1	SPI1_MOS
	3	PA2		ADC_IN4	GT_CH2	SPI1_MISO
	4	PA3		ADC_IN5	GT_CH3	SPI1_SEL

- 做为 PWM 输出模式时, 可设置 PWM 的频率, 占空比。
GT_CH0, GT_CH1, GT_CH3 为灰色是因为没在 AFIO 页面里选中。

● Output PWM Mode





Frequency: Hz

CH0 On Duty: % CH2 On Duty: %

CH1 On Duty: % CH3 On Duty: %

- 做为输入捕捉模式时，可设置捕提高电平宽度，低电平宽度，或者周期。
因为 GT_CH2 的特殊性，使能 CH2 可以同时使用 CH2, CH3 的捕捉功能，详细请参考 MCU 的 User Manual。

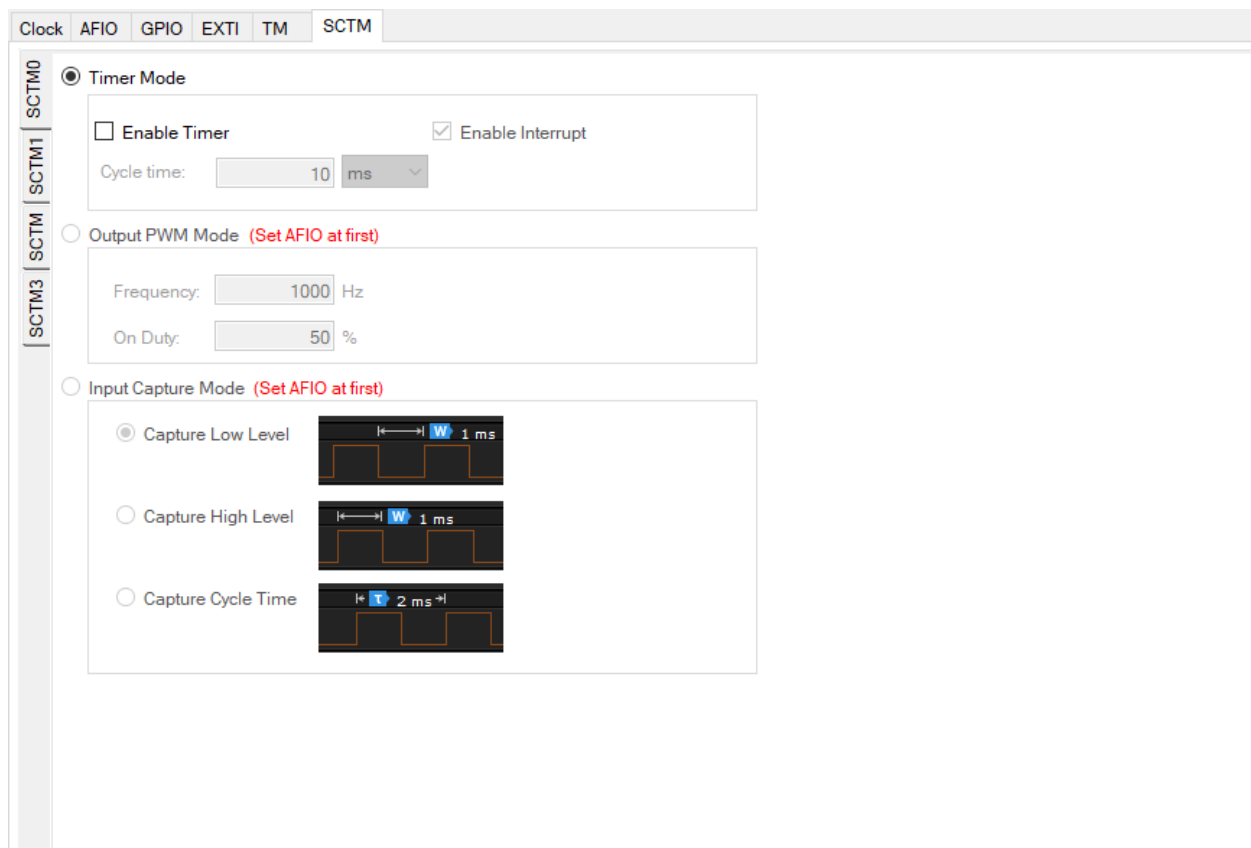
● Input Capture Mode

<input type="checkbox"/> Enable CH0	TI0	Capture Low Level	
<input type="checkbox"/> Enable CH1	TI1	Capture Low Level	
<input type="checkbox"/> Enable CH2	TI2	Capture Low Level	
<input type="checkbox"/> Enable CH3	TI3	Capture Low Level	

- MCTM 除了类似 GPTM 的功能之外，还有更多功能，本软件暂只提供与 GPTM 类似的设置，其它更高级的设置请参考 User Manual 后手动修改 source code 实现。
MCTM 类似的设置项参考 GPTM。
- PWM 与 GPTM 相比，删减了捕捉功能，其它功能类似 GPTM。
PWM 相关设置项参考 GPTM。

8. “SCTM”页面的设置

SCTM 并不是每个型号的 MCU 都有，若有，则会显示本页面



- SCTM 与 GPTM 的差异之处是 SCTM 只有一个通道，而 GPTM 有 4 个通道。
- SCTM 同样有基础定时器，PWM 输出，输入捕捉功能。后两者需要先在 AFIO 页面选中，才能在本页面选择。
- SCTM 相关设置项参考 GPTM。

9. “ADC/DAC”页面的设置

本页面需要先在 AFIO 页面里选择了 ADC, 或者 DAC, 才会显示出来。

9.1 ADC 的说明:

先在 AFIO 页面选中 ADC_IN5, ADC_IN6,

Clock	AFIO	GPIO	EXTI	TM	SCTM	ADC
	PIN NO.	AF0 Default	AF1 GPIO	AF2 ADC	AF4 TM	
	1	PA0		ADC_IN2	GT_CH0	S
	2	PA1		ADC_IN3	GT_CH1	S
	3	PA2		ADC_IN4	GT_CH2	S
	4	PA3		ADC_IN5	GT_CH3	S
▶	5	PA4		ADC_IN6	GT_CH0	S
	6	PA5		ADC_IN7	GT_CH1	S

Clock AFIO GPIO EXTI TM SCTM ADC

ADC

ADC Group1

Conversion Sequence Select 0:

ADC_IN5

Conversion Sequence Select 1:

ADC_IN6

Conversion Sequence Select 2:

Conversion Sequence Select 3:

Conversion Sequence Select 4:

Conversion Sequence Select 5:

Conversion Sequence Select 6:

Conversion Sequence Select 7:

Input Sampling Time[0~255]:

0

ADC Clock Prescaler:

CKCU_ADCPRE_DIV64

Conversion Mode:

ONE_SHOT_MODE

☐ Enable Single EOC Interrupt

☐ Enable Subgroup EOC Interrupt

☒ Enable Cycle EOC Interrupt

- ADC 组成员会根据 AFIO 页面选择的 ADC 通道，自动填好。
- 可以设置采样时间，延长采样时间可以让采样值更准确，不过缺点是使得 ADC 转换总时间变长，所以需要合理设置。
- 可以设置 ADC 的时钟预分频，设置转换模式，设置转换完成的中断类型。
- 对于普通应用，本页面基本无需修改。

9.2 DAC 的说明:

先在 AFIO 页面选中 DAC0_OUT,

Clock	AFIO	GPIO	EXTI	TM	SCTM	ADC/DAC
	PIN NO.	AF0 Default	AF1 GPIO	AF2 ADC		
▶	71	VSS_3				
	72	PE3				COU
	73	PC1				CN0
	74	PC2				CP0
	75	PC3		DAC0_OUT		COU
	76	PB6				CN1
	77	PB7				CP1
	78	PB8		DAC1_OUT		COU
	79	VDDA				

Clock

AFIO

GPIO

EXTI

TM

SCTM

ADC/DAC

DAC

DAC

Conversion mode

ASYNC_MODE

DAC0 Reference

DAC_REFERENCE_VDDA

DAC0 Resolution

DAC_RESOLUTION_12BIT

DAC1 Reference

DAC_REFERENCE_VDDA

DAC1 Resolution

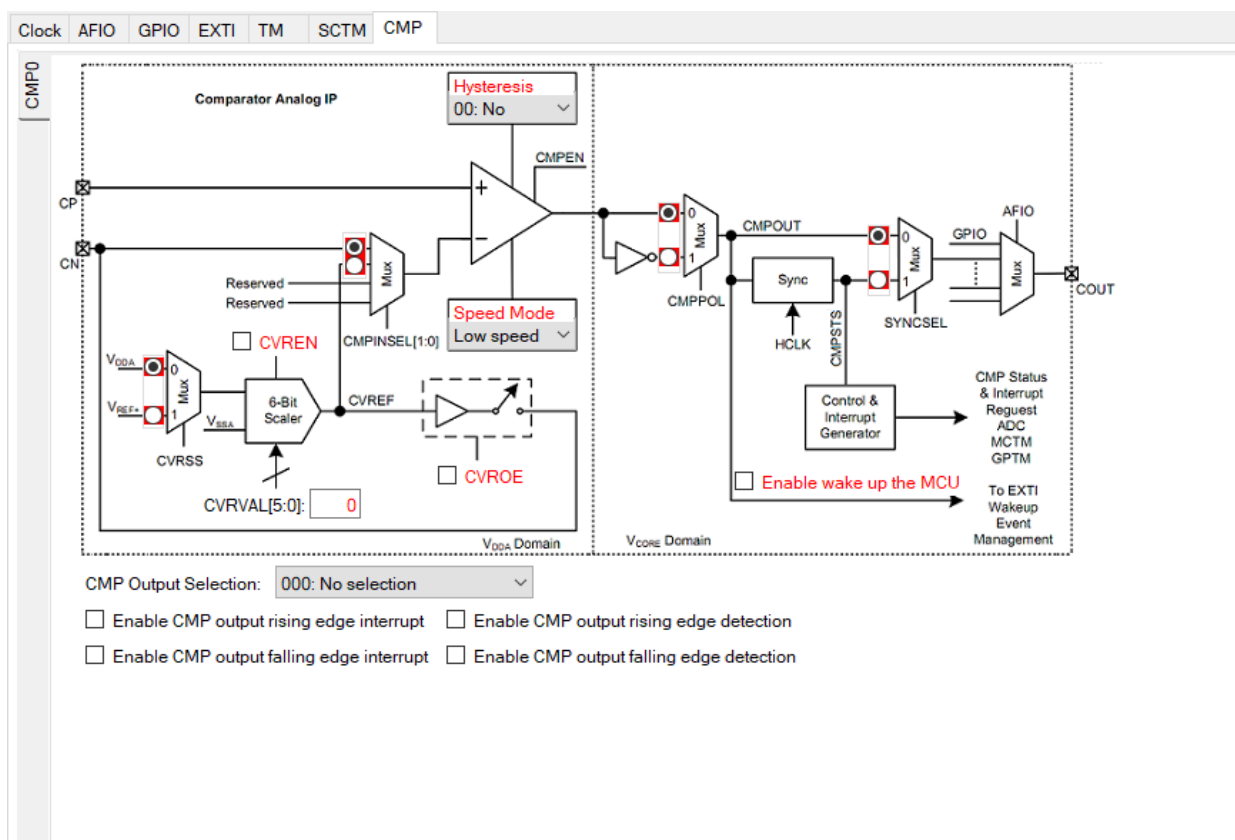
DAC_RESOLUTION_12BIT

☒ Enable Interrupt

- 不是所有型号的 MCU 都具有 DAC 功能，具体请查阅 User Manual。
- DAC 可以设置转换模式为异步或者同步模式，参考电压 VDDA 或者 VREF，转换精度 12bit 或者 8bit，转换完成是否触发中断。
- 若对应的通道在 AFIO 页面被选中，该通道才是可选择状态，否则为灰色，如上图 DAC1。

10. “CMP”页面的设置

若所选 MCU 具有 CMP 功能，并且 AFIO 页面选中了 CN,CP,或者 COUT，则 CMP 页面会显示出来。



- 本页面左侧显示所有 AFIO 页面选中的的 CMP。
- 图形化的选择，按需点击选择功能。
- 可设置迟滞 hysteresis，高速或低速模式，CN 的参考源来自外部或者内部；内部参考源可以设置是否分压，分压低级；COUT 输出是否触发中断，是否唤醒 MCU 等等。

11. “SPI”页面的设置

SPI 是很常用的功能。在 AFIO 页面里选中任意 SPI 之后，SPI 页面就会显示出来。

Clock	AFIO	GPIO	EXTI	TM	SCTM	SPI		
	PIN NO.	AF0 Default	AF1 GPIO	AF2 ADC	AF3 CMP	AF4 TM	AF5 SPI	AF UA
	1	PA0		ADC_IN0		GT_CH0	SPI1_SCK	USR_F
	2	PA1		ADC_IN1		GT_CH1	SPI1_MOSI	USR_C
	3	PA2		ADC_IN2		GT_CH2	SPI1_MISO	USR_T
	4	PA3		ADC_IN3		GT_CH3	SPI1_SEL	USR_F
	5	PA4		ADC_IN4		GT_CH0	SPI0_SCK	USR_T
	6	PA5		ADC_IN5		GT_CH1	SPI0_MOSI	USR_F
	7	PA6		ADC_IN6		GT_CH2	SPI0_MISO	USR_F
▶	8	PA7		ADC_IN7		GT_CH3	SPI0_SEL	USR_C
	9	PD4		ADC_IN8				UR1_T

Clock

AFIO

GPIO

EXTI

TM

SCTM

SPI

SPI0

Mode:

SPI_MASTER

Data Length:

SPI_DATALENGTH_8

SEL Mode:

SPI_SEL_SOFTWARE

SEL Polarity:

SPI_SELPOLARITY_LOW

First Bit:

SPI_FIRSTBIT_MSB

CPOL:

SPI_CPOL_LOW

CPHA:

SPI_CPHA_FIRST

Clock Prescaler:

0

×2 + 2

☒ Enable FIFO

RxFIFO Trigger Level:

4

TxFIFO Trigger Level:

4

☐ Enable SPI IRQ

SEL (SELAP=0)

SEL (SELAP=1)

SCK

MOSI

MISO

data sampled

- 本页面左侧显示所有 AFIO 页面选中的 SPI。
- SPI 可以设置主从模式，数据长度，SEL 模式，MSB/LSB，SPI 的 4 种模式以及时钟预分频，FIFO 深度，是否使能中断。
- 右侧会根据选择的选项，把波形图形化，直观的反馈选择是否正确。

12. “UART/USART”页面的设置

UART/USART 也是很常用的功能。在 AFIO 页面里选中任意 UART/USART 之后，本页面就会显示出来。

	Clock	AFIO	GPIO	EXTI	TM	SCTM	UART/USART		
	PIN NO.	AF0 Default	AF1 GPIO	AF2 ADC	AF3 CMP	AF4 TM	AF5 SPI	AF6 UART	AF7 I2C
	1	PA0		ADC_IN0		GT_CH0	SPI1_SCK	USR_RTS	I2C1_SCL
	2	PA1		ADC_IN1		GT_CH1	SPI1_MOSI	USR_CTS	I2C1_SD
	3	PA2		ADC_IN2		GT_CH2	SPI1_MISO	USR_TX	
▶	4	PA3		ADC_IN3		GT_CH3	SPI1_SEL	USR_RX	
	5	PA4		ADC_IN4		GT_CH0	SPI0_SCK	USR_TX	I2C0_SCL
	6	PA5		ADC_IN5		GT_CH1	SPI0_MOSI	USR_RX	I2C0_SD
	7	PA6		ADC_IN6		GT_CH2	SPI0_MISO	USR_RTS	

Clock AFIO GPIO EXTI TM SCTM **UART/USART**

USART0

Mode: USART_MODE_NORMAL ☐ As Retarget port (printf)
Baud Rate: 115200 bps
Data Length: USART_WORDLENGTH_8B
Stop bits: USART_STOPBITS_1
Parity No: USART_PARITY_NO
☒ Enable FIFO
Rx FIFO Trigger Level: 4
Tx FIFO Trigger Level: 4
☒ Enable Timeout IRQ
☒ Enable IRQ

- 本页面左侧显示所有 AFIO 页面选中的 UART/USART。
- 如果选中某个 As Retarget port 做为打印的串口，则其它 UART/USART 就不能再做打印串口。
- USART 可设置成普通串口模式，或者 IRDA, 485，或者同步模式，即把 USART 当成 SPI 使用，UART 没有这些功能。。
- USART 有 FIFO 的设置以及 timeout 中断设置，UART 没有这些功能。
- 可设置波特率，数据长度，停止位数，校验位。
- 同步模式有更多的类似 SPI 选项，如下图。

SYNCHRONOUS Mode Config

First Bit: USART_MSB_FIRST
CPOL: USART_SYN_CLOCK_POLARITY_LOW
CPHA: USART_SYN_CLOCK_PHASE_FIRST
SEL: PA0

13. “I2C”页面的设置

I2C 也是很常用的功能。在 AFIO 页面里选中任意 I2C 之后，本页面就会显示出来。

Clock	AFIO	GPIO	EXTI	TM	SCTM	I2C				
	PIN NO.	AF0 Default	AF1 GPIO	AF2 ADC	AF3 CMP	AF4 TM	AF5 SPI	AF6 UART	AF7 I2C	AF8
	1	PA0		ADC_IN0		GT_CH0	SPI1_SCK	USR_RTS	I2C1_SCL	SCI0_C
	2	PA1		ADC_IN1		GT_CH1	SPI1_MOSI	USR_CTS	I2C1_SDA	SCI0_C
	3	PA2		ADC_IN2		GT_CH2	SPI1_MISO	USR_TX		
	4	PA3		ADC_IN3		GT_CH3	SPI1_SEL	USR_RX		
	5	PA4		ADC_IN4		GT_CH0	SPI0_SCK	USR_TX	I2C0_SCL	SCI1_C
▶	6	PA5		ADC_IN5		GT_CH1	SPI0_MOSI	USR_RX	I2C0_SDA	SCI1_C
	7	PA6		ADC_IN6		GT_CH2	SPI0_MISO	USR_RTS		SCI1_C
	8	PA7		ADC_IN7		GT_CH3	SPI0_SEL	USR_CTS		

Clock AFIO GPIO EXTI TM SCTM I2C

I2C0

Mode: Master
Addressing Mode: 7bit
Speed: 100000 bps
Address: 1 (DEC)

- 本页面左侧显示所有 AFIO 页面选中的 I2C。
- 可以设置主从模式，地址 7bit 或者 10bit 模式；主模式可以设置速度，从模式可以设置被访问时的设备地址。
- 设置很简单，最终输出的代码不光会包含读与写操作，还有借助 BFTM 的纠错机制，防止干扰信号或者连接反复断开/接上等情况下的 I2C 锁死状况的出现。

14. “SCI”页面的设置

在 AFIO 页面里选中任意 SCI 之后，本页面就会显示出来。

Clock	AFIO	GPIO	EXTI	TM	SCTM	SCI					
	PIN NO.	AF0 Default	AF1 GPIO	AF2 ADC	AF3 CMP	AF4 TM	AF5 SPI	AF6 UART	AF7 I2C	AF8 SCI	I2S_
	1	PA0		ADC_IN0		GT_CH0	SPI1_SCK	USR_RTS	I2C1_SCL	SCI0_CLK	I2S_
	2	PA1		ADC_IN1		GT_CH1	SPI1_MOSI	USR_CTS	I2C1_SDA	SCI0_DIO	I2S_
	3	PA2		ADC_IN2		GT_CH2	SPI1_MISO	USR_TX			I2S_
	4	PA3		ADC_IN3		GT_CH3	SPI1_SEL	USR_RX			I2S_
	5	PA4		ADC_IN4		GT_CH0	SPI0_SCK	USR_TX	I2C0_SCL	SCI1_CLK	
	6	PA5		ADC_IN5		GT_CH1	SPI0_MOSI	USR_RX	I2C0_SDA	SCI1_DIO	
▶	7	PA6		ADC_IN6		GT_CH2	SPI0_MISO	USR_RTS		SCI1_DET	
	8	PA7		ADC_IN7		GT_CH3	SPI0_SEL	USR_CTS			I2S_

SCI1

Mode: MODE_SCI

CMD GPIO: PA2

Retry: RETRY_NO

RESET GPIO: PA3

Convention: CONVENTION_DIRECT

Card Polarity: CARDPOLARITY_LOW

Clock Prescale: CLKPRESCALER_12

GuardTime: 16

ETU: 372

Compensation: COMPENSATION_DISABLE

ClockMode: CLK_HARDWARE

- 本页面左侧显示所有 AFIO 页面选中的 SCI。
- 可以设置模式，Retry，Convention，Card Polarity，Clock Prescale，GuardTime，ETU，Compensation，ClockMode。
- CMD GPIO，RESET GPIO 会列出所有可用的 GPIO，供选择。

15. “EBI”页面的设置

15.1 EBI 基础设置：

在 AFIO 页面里选中任意 EBI 之后，本页面就会显示出来。

Clock	AFIO	GPIO	EXTI	TM	SCTM	EBI					
	AF3 CMP	AF4 TM	AF5 SPI	AF6 UART	AF7 I2C	AF8 SCI	AF9 EBI	AF10 I2S	AF13 SCTM	AF15 Other	
		MT_CH0	SPI1_SEL	USR1_TX	I2C1_SCL	SCI0_CLK	EBI_AD0				
		MT_CH0N	SPI1_SCK	USR1_RX	I2C1_SDA	SCI0_DIO	EBI_AD1		SCTM1		
		MT_CH1	SPI1_MOSI	USR0_TX	I2C0_SCL		EBI_AD2				

因为 EBI 脚位固定，并且脚位较多，第一次选择 EBI 功能会有如下的辅助选择窗口弹出：

Auto Select

☒ EBI AD0 ~ AD7
☐ EBI AD0 ~ AD15

☒ EBI CS0
☐ EBI CS1
☐ EBI CS2
☐ EBI CS3

OK

Cancel

Note: need select EBI_Ax manually

例如勾选 AD0~AD7 + EBI CS0，点击 OK，则 AFIO 页面中相关位置会自动被选中。

Clock AFIO GPIO EXTI TM SCTM EBI

EBI

Bank: 0

Mode: D8A8

Idle Cycle: ENABLE

Chip Select Polarity: LOW

Address Latch Polarity: LOW

Write Enable Polarity: LOW

Read Enable Polarity: LOW

Idle Cycle Time: 0

Address Setup Time: 0

Address Hold Time: 0

Write Setup Time: 2

Write Strobe Time: 2

Write Hold Time: 1

Read Setup Time: 2

Read Strobe Time: 5

Read Hold Time: 1

TFT Driver IC

Null

Reset + Backlight pin

Reset pin: PA0

☒ BL pin PA1

☐ Enable LCD factory

SPI Flash: (Set SPI at first)

- EBI 的选项很多，如果不太清楚具体用途，就不要修改。

15.2 驱动 TFT 屏：

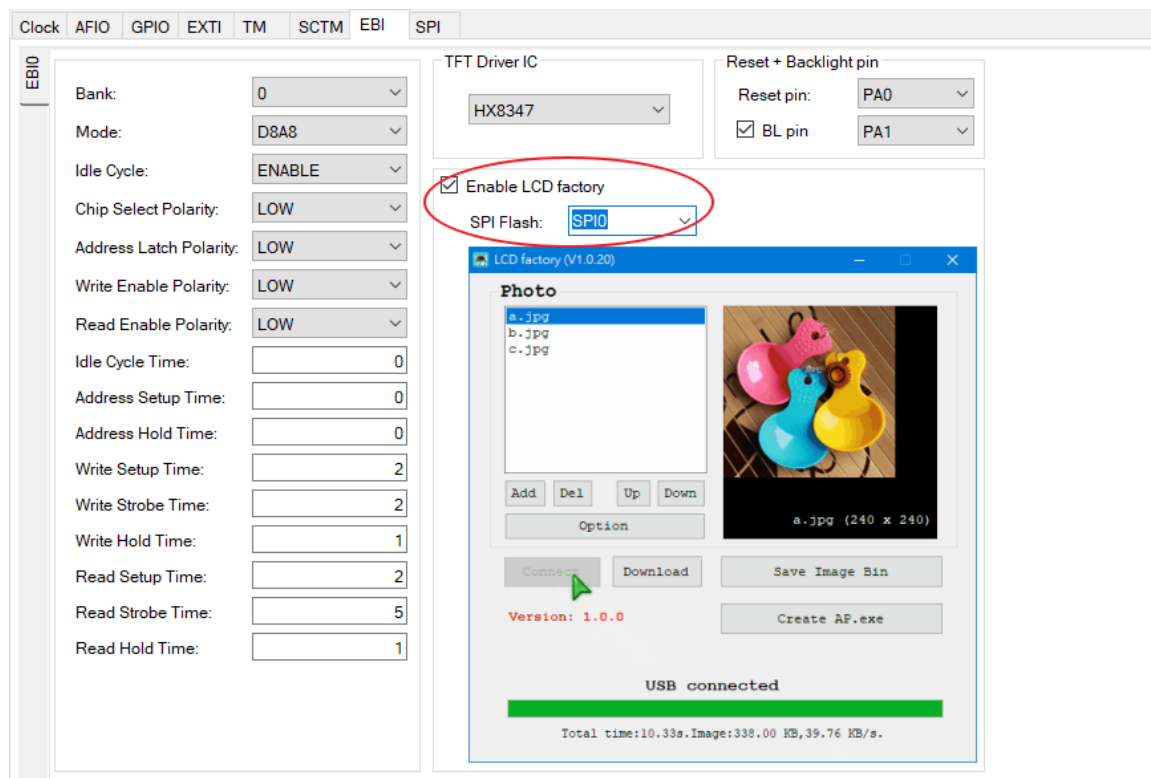
- 如果需要用 EBI 来驱动 TFT 屏，则设置右侧的 TFT 驱动 IC 的类型，屏的复位脚，背光控制脚。背光也可能是 PWM 信号，那就请用 TM 或者 SCTM 页面设置一个 PWM。到此处为止，输出的代码可以让屏显示一个纯色，例如全红色。

The screenshot shows the 'EBI' configuration page in the HT32init tool. The page is divided into several sections:

- EBI Settings (Left Panel):**
 - Bank: 0
 - Mode: D8A8
 - Idle Cycle: ENABLE
 - Chip Select Polarity: LOW
 - Address Latch Polarity: LOW
 - Write Enable Polarity: LOW
 - Read Enable Polarity: LOW
 - Idle Cycle Time: 0
 - Address Setup Time: 0
 - Address Hold Time: 0
 - Write Setup Time: 2
 - Write Strobe Time: 2
 - Write Hold Time: 1
 - Read Setup Time: 2
 - Read Strobe Time: 5
 - Read Hold Time: 1
- TFT Driver IC (Top Right):**
 - TFT Driver IC: HX8347
- Reset + Backlight pin (Top Right):**
 - Reset pin: PA0
 - ☒ BL pin: PA1
- Enable LCD factory (Bottom Right):**
 - ☐ Enable LCD factory
 - SPI Flash: (Set SPI at first)

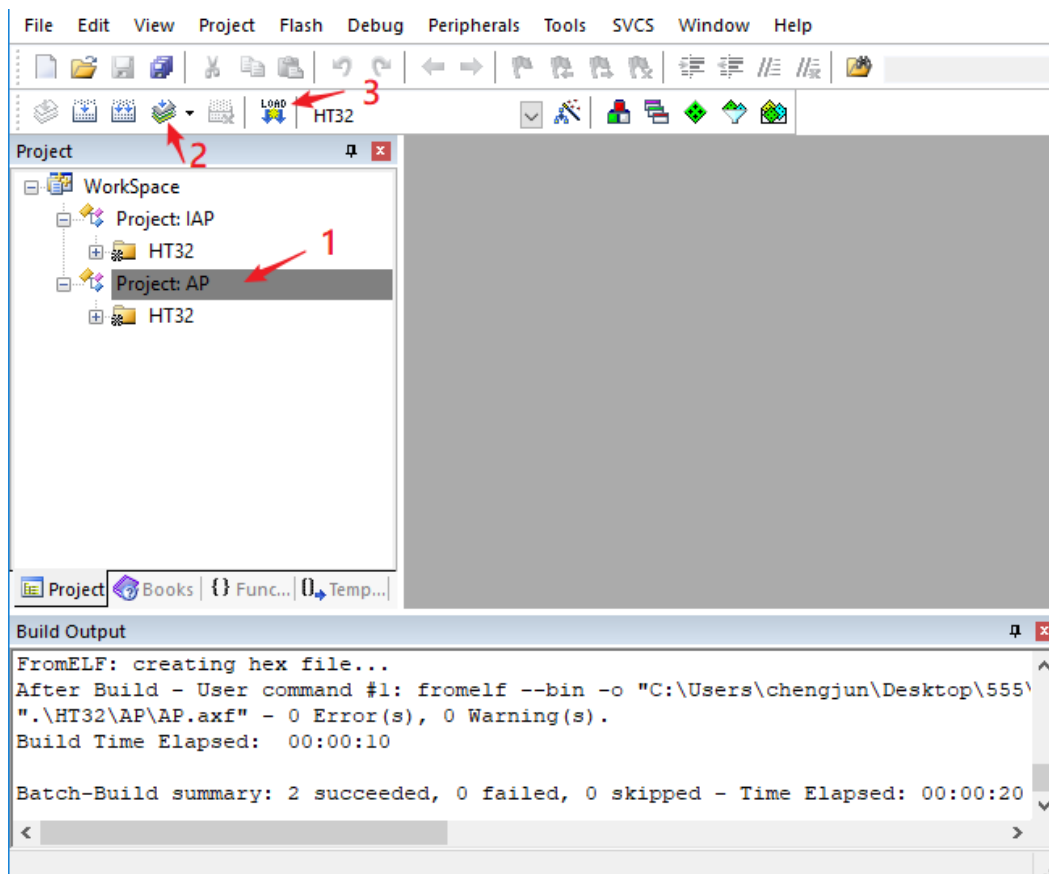
15.3 LCD 平台：

- 如果还需要显示图片，更新图片等功能，需要使用合泰提供的 LCD 平台来协助。LCD 平台是一套包含上位机程序，通过 USB 接口更新图片的整套解决方案。因为相对 MCU ROM 而言图片数据比较大，所以都保存在外部 SPI flash 内。刷图时从 SPI flash 读出图片数据送到 TFT 屏上显示。
- 先在 AFIO 页面选中一个 SPI 用于连接外部的 SPI flash，再在本页面里使能 LCD factory 并选择 SPI，就激活了 LCD 平台的功能。

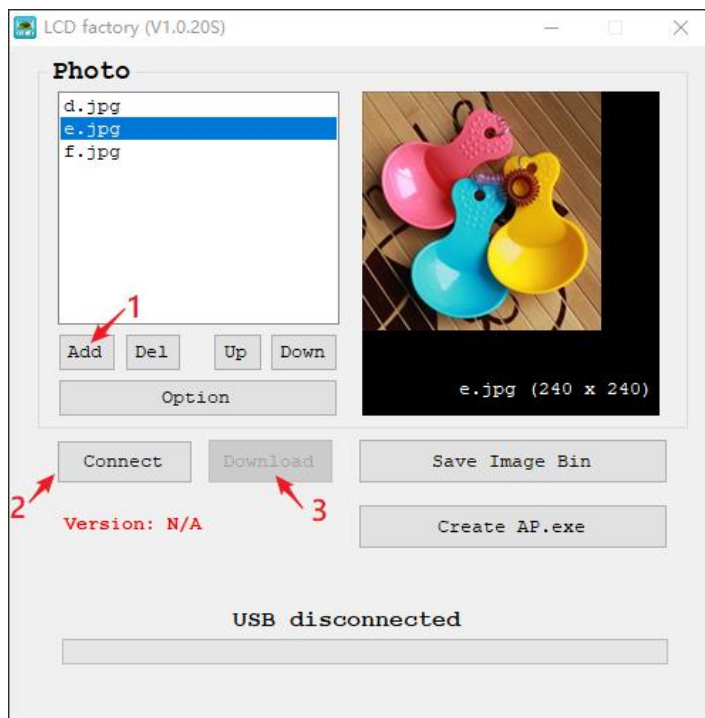


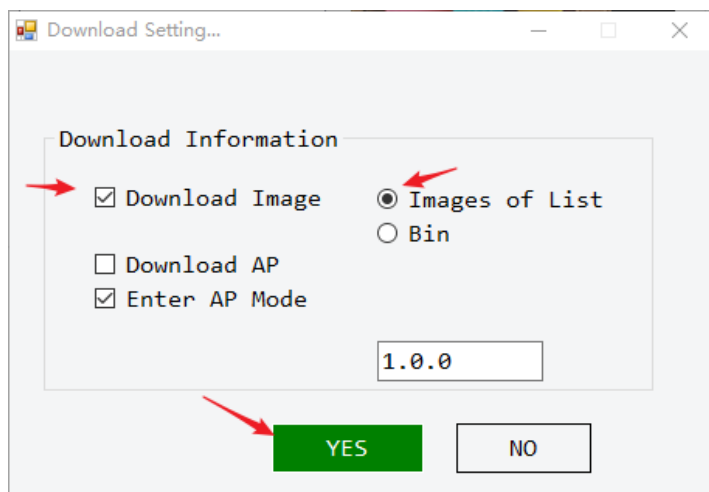
- 界面有动画显示操作流程
- 输出的文件包含有上位机程序，和固件的源代码。
- 支持 8bit EBI, 16bit EBI, 支持图片压缩功能以尽量减小使用的 SPI flash 大小。
- 详细使用步骤：
 - 生成的工程包含 IAP, AP 两个工程，点击 Multiproject 可打开两个工程，如下图：

名称	修改日期	类型	大小
AP.uvoptx	2019-12-25 11:39	UVOPTX 文件	4 KB
AP.uvprojx	2019-12-25 11:39	UVPROJX 文件	27 KB
ht32_op.s	2019-08-02 20:14	Assembler Source	19 KB
HT32F5xxx_01_DebugSupport.ini	2019-07-08 12:58	配置设置	6 KB
iap.s	2019-12-25 11:39	Assembler Source	1 KB
IAP.uvprojx	2019-12-25 11:39	UVPROJX 文件	24 KB
image_Decompress.lib	2019-07-26 16:27	Altium Library	182 KB
LCD_Platform.lib	2019-07-26 17:10	Altium Library	174 KB
LCD_Platform.uvmpw	2019-12-25 11:39	UVMPW 文件	1 KB
linker.lin	2019-12-25 11:39	LIN 文件	2 KB
startup_ht32f5xxx_01.s	2019-12-25 11:39	Assembler Source	21 KB
version.s	2019-12-25 11:39	Assembler Source	1 KB
version.txt	2019-12-25 11:39	文本文档	1 KB



- 设置 AP 为 Active Project
- Batch Build IAP 和 AP 两个工程
- 下载到 MCU
- 用 USB 线连接 MCU 到电脑
- 打开 LCD factory.exe, Add 选择图片, connect 连接, download 下载





下载成功后就可以看到 TFT 上图片定时刷新了。

如果失败，检查：

- ✧ 屏的 EBI 设定是 8bit 还是 16bit；
- ✧ 如果屏有 OE 脚，检查 OE 脚的设置；
- ✧ 检查 SPI 各个脚位是否设置错误；
- ✧ 检查 RESET 脚是否设置错误；
- ✧ 检查背光控制脚是否设置错误；

16. “I2S”页面的设置

在 AFIO 页面里选中任意 I2S 之后，本页面就会显示出来。

Clock	AFIO	GPIO	EXTI	TM	SCTM	I2S					
	AF3 CMP										
		GT1_CH0	SPI1_SCK	USR0_RTS	I2C1_SCL	SCI0_CLK		I2S_WS			
		GT1_CH1	SPI1_MOSI	USR0_CTS	I2C1_SDA	SCI0_DIO		I2S_BCLK			
		GT1_CH2	SPI1_MISO	USR0_TX				I2S_SDO			
		GT1_CH3	SPI1_SEL	USR0_RX				I2S_SDI			
		GT0_CH0	SPI0_SCK	USR1_TX	I2C0_SCL	SCI1_CLK					
		GT0_CH1	SPI0_MOSI	USR1_RX	I2C0_SDA	SCI1_DIO					
		GT0_CH2	SPI0_MISO	USR1_RTS		SCI1_DET					
		GT0_CH3	SPI0_SEL	USR1_CTS				I2S_MCLK			

Clock

AFIO

GPIO

EXTI

TM

SCTM

I2S

I2S0

Mode:

MASTER_TX

Format:

I2S_JUSTIFIED_STEREO

Word Width:

WORDWIDTH_16

Mclk Out:

ENABLE

Mclk Inv:

DISABLE

Bclk Inv:

DISABLE

X_Div:

8

Y_Div:

125

N_Div:

0

MCLK = 1536000

BCLK = 1536000

Sample rate = 48000

☒ Recommend X/Y/N for me

Target sample rate:

48000

- 可以设置主从各种模式，数据格式，数据宽度，MCLK，BCLK。
- 通过设置 X, Y, N 的值可以设定采样率。
- 如果是常用的采样率，可以通过选择推荐值，自动计算出 X, Y, N。

17. “SDIO”页面的设置

在 AFIO 页面里选中任意 SDIO 之后，本页面就会显示出来。

Clock	AFIO	GPIO	EXTI	TM	SDIO						
	AF3 CMP	AF4 TM	AF5 SPI	AF6 UART	AF7 I2C	AF8 SCI	AF9 EBI	AF10 I2S	AF11 SDIO	AF15 Other	
		GT1_CH0	SPI1_SCK	USR0_RTS	I2C1_SCL	SCI0_CLK		I2S_WS			
		GT1_CH1	SPI1_MOSI	USR0_CTS	I2C1_SDA	SCI0_DIO		I2S_BCLK	SD_DAT1		
		GT1_CH2	SPI1_MISO	USR0_TX				I2S_SDO	SD_DAT2		
		GT1_CH3	SPI1_SEL	USR0_RX				I2S_SDI	SD_DAT3		
		GT0_CH0	SPI0_SCK	USR1_TX	I2C0_SCL	SCI1_CLK			SD_CMD		
		GT0_CH1	SPI0_MOSI	USR1_RX	I2C0_SDA	SCI1_DIO			SD_CLK		
		GT0_CH2	SPI0_MISO	USR1_RTS		SCI1_DET			SD_DAT0		

Clock AFIO GPIO EXTI TM SDIO

SDIO

☒ Initial SD card
☒ Add FatFs over SDIO

- 此处的 SDIO 主要用来读写 SD/TF 卡, 所以选项很简化。
- 可以设置是否增加 FatFs 文件系统读写 SD/TF 卡。

18. “CSIF”页面的设置

在 AFIO 页面里选中任意 CSIF 之后，本页面就会显示出来。

Clock	AFIO	GPIO	EXTI	TM	CSIF					
	AF4 TM	AF5 SPI	AF6 UART	AF7 I2C	AF8 SCI	AF9 EBI	AF10 I2S	AF11 SDIO	AF12 CSIF	AF15 Other
	GT1_CH0	SPI1_SCK	USR0_RTS	I2C1_SCL	SCI0_CLK		I2S_WS		CSIF_D0	
	GT1_CH1	SPI1_MOSI	USR0_CTS	I2C1_SDA	SCI0_DIO		I2S_BCLK	SD_DAT1	CSIF_D1	
	GT1_CH2	SPI1_MISO	USR0_TX				I2S_SDO	SD_DAT2	CSIF_D2	
	GT1_CH3	SPI1_SEL	USR0_RX				I2S_SDI	SD_DAT3	CSIF_D3	
	GT0_CH0	SPI0_SCK	USR1_TX	I2C0_SCL	SCI1_CLK			SD_CMD	CSIF_D4	
	GT0_CH1	SPI0_MOSI	USR1_RX	I2C0_SDA	SCI1_DIO			SD_CLK	CSIF_D5	
	GT0_CH2	SPI0_MISO	USR1_RTS		SCI1_DET			SD_DAT0	CSIF_D6	
	GT0_CH3	SPI0_SEL	USR1_CTS				I2S_MCLK		CSIF_D7	
		SPI1_SEL	USR0_RTS						CSIF_HSY...	
		SPI1_SCK	USR0_CTS						CSIF_VSYNC	
		SPI1_MOSI	USR0_TX						CSIF_PCK	
		SPI1_MISO	USR0_RX						CSIF_MCK	

Clock AFIO GPIO EXTI TM CSIF I2C

CSIF0

☒ Initial CSIF
☒ Read GT5110 demo
Select I2C:
Reset pin:

- 默认使能初始化 CSIF 接口。
- 可以设置是否初始化 GT5110 摄像头模组。初始化模组还需要一组 I2C 和复位脚。

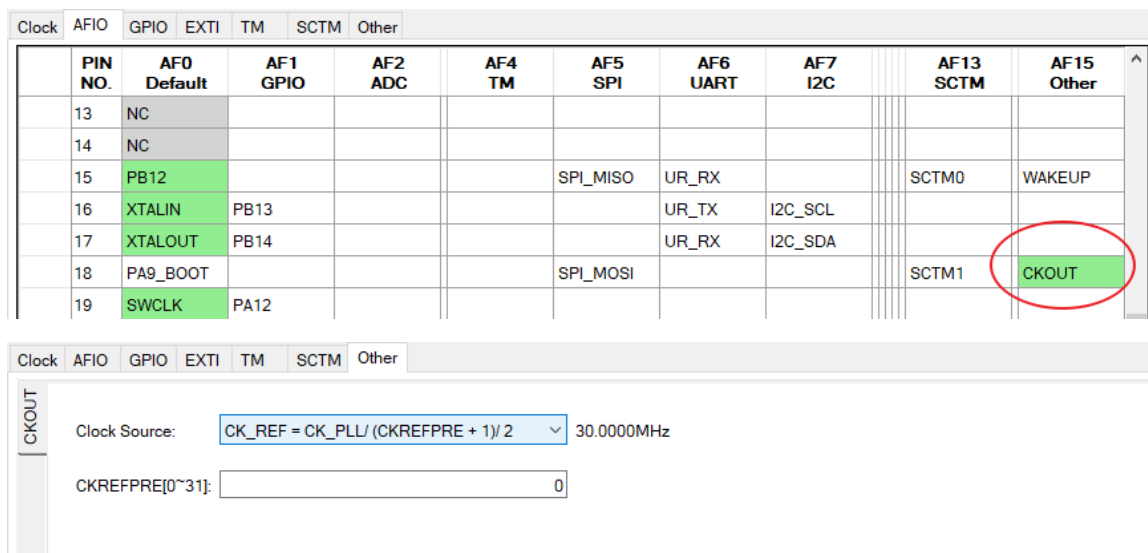
- 可以设置 MonoLCD 的时钟源，预分频，分频器，占空比，偏置，电压源，波形，充电泵电压以及中断使能位。
- 具体如何设置需要按选用的 LCD 屏而定。

20. “Other”页面的设置

本页面包含有 CKOUT, PDMA, UART IAP, USB IAP 的相关设置。

在 AFIO 页面里选中 CKOUT 之后，或者菜单栏 Tools 里选择 Add UART IAP, Add USB IAP, 或者 Add PDMA 本页面就会显示出来。

20.1 CKOUT 设置：

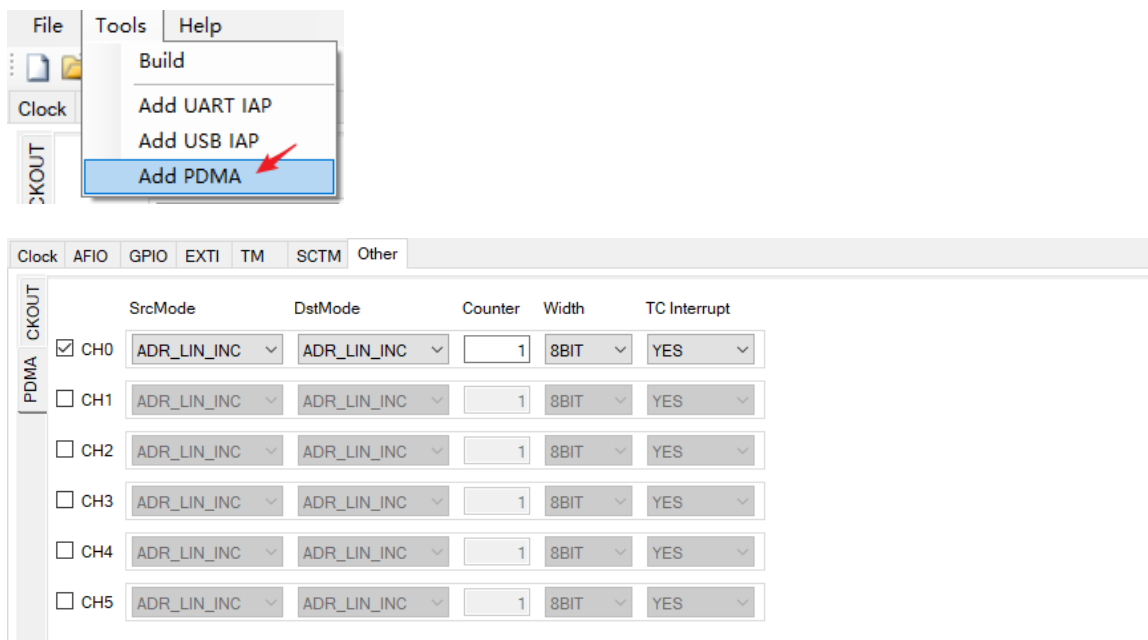


PIN NO.	AF0 Default	AF1 GPIO	AF2 ADC	AF4 TM	AF5 SPI	AF6 UART	AF7 I2C	AF13 SCTM	AF15 Other
13	NC								
14	NC								
15	PB12				SPI_MISO	UR_RX		SCTM0	WAKEUP
16	XTALIN	PB13				UR_TX	I2C_SCL		
17	XTALOUT	PB14				UR_RX	I2C_SDA		
18	PA9_BOOT				SPI_MOSI			SCTM1	CKOUT
19	SWCLK	PA12							

Clock Source: $CK_REF = CK_PLL / (CKREFPRE + 1) / 2$ 30.0000MHz
 CKREFPRE[0~31]: 0

- 在 AFIO 页面里选中 CKOUT 之后，本页面可以选择 CKOUT 的 7 种时钟源之一，并会实时显示出最终输出的时钟频率值。

20.2 PDMA 设置：



File Tools Help

Build

Add UART IAP

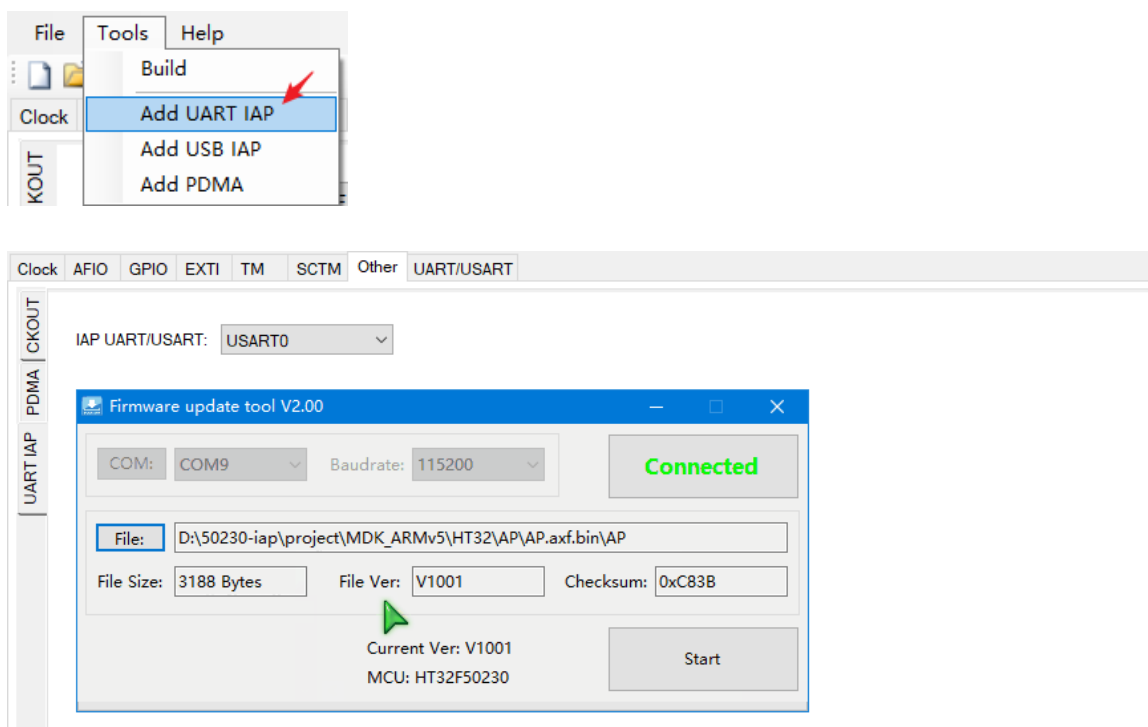
Add USB IAP

Add PDMA

	SrcMode	DstMode	Counter	Width	TC Interrupt
<input checked="" type="checkbox"/> CH0	ADR_LIN_INC	ADR_LIN_INC	1	8BIT	YES
<input type="checkbox"/> CH1	ADR_LIN_INC	ADR_LIN_INC	1	8BIT	YES
<input type="checkbox"/> CH2	ADR_LIN_INC	ADR_LIN_INC	1	8BIT	YES
<input type="checkbox"/> CH3	ADR_LIN_INC	ADR_LIN_INC	1	8BIT	YES
<input type="checkbox"/> CH4	ADR_LIN_INC	ADR_LIN_INC	1	8BIT	YES
<input type="checkbox"/> CH5	ADR_LIN_INC	ADR_LIN_INC	1	8BIT	YES

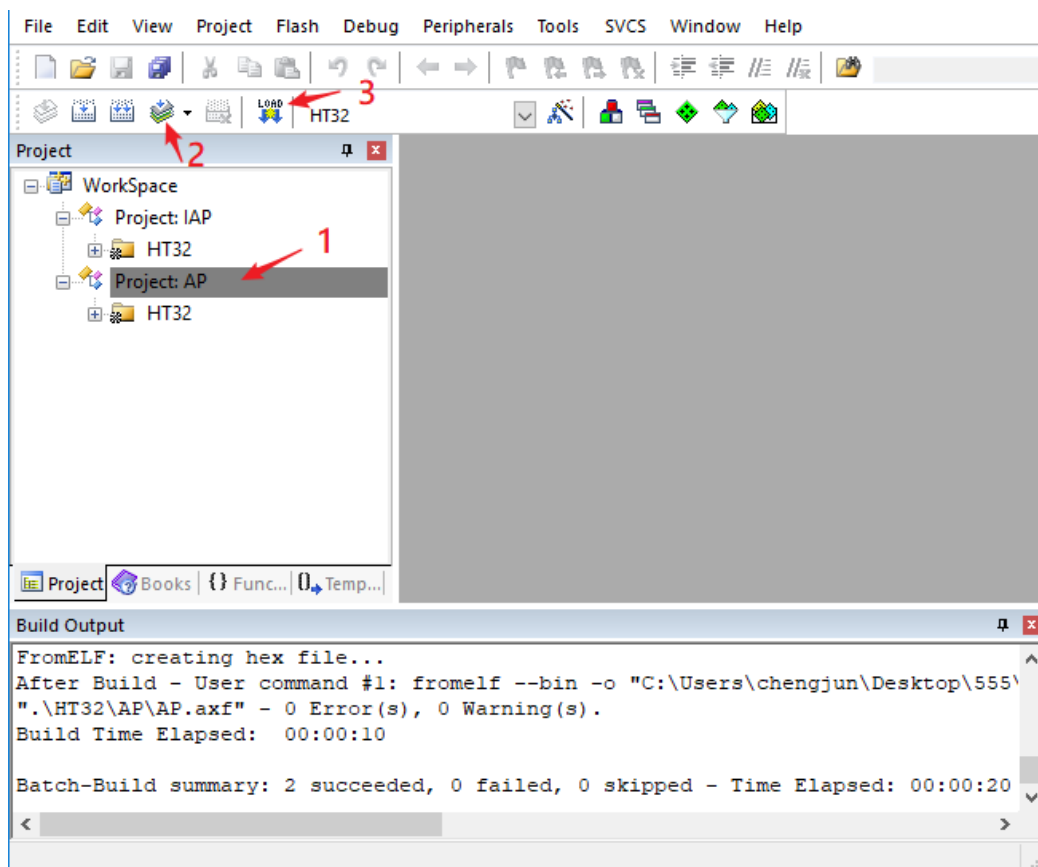
- 如果选用的 MCU 支持 PDMA, 则可以通过菜单栏 Tools→Add PDMA, 激活 PDMA 的设置。
- PDMA 可以独立使能各个通道, 每个通道可以设置源地址的模式, 目标地址的模式, 传输的数量, 数据宽度, 以及是否在传输结束时触发中断。

20.3 UART IAP 设置:

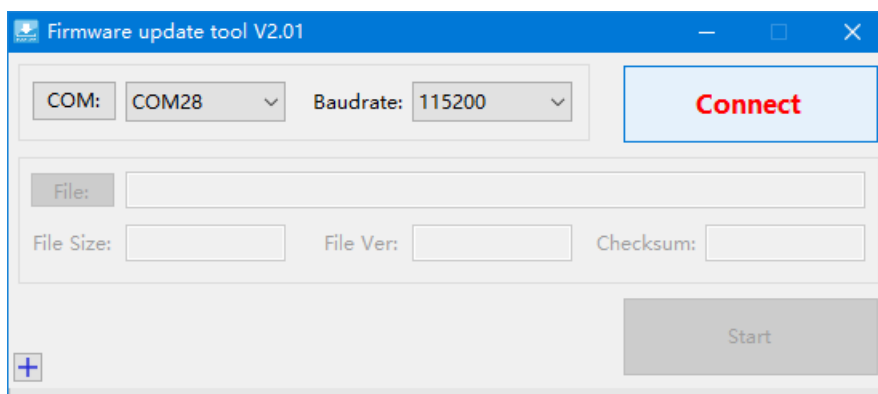


- UART IAP 是一套完整的解决方案，包括上位机程序和固件源程序。
- 先使能一组 UART，就可以通过菜单栏 Tools→Add UART IAP 激活此功能。
- 本页面的设置只有选择哪组 UART，其它都无需设置。
- 详细使用步骤：
 - 生成的工程包含 IAP, AP 两个工程，点击 Multiproject 可以打开两个工程，如下图：

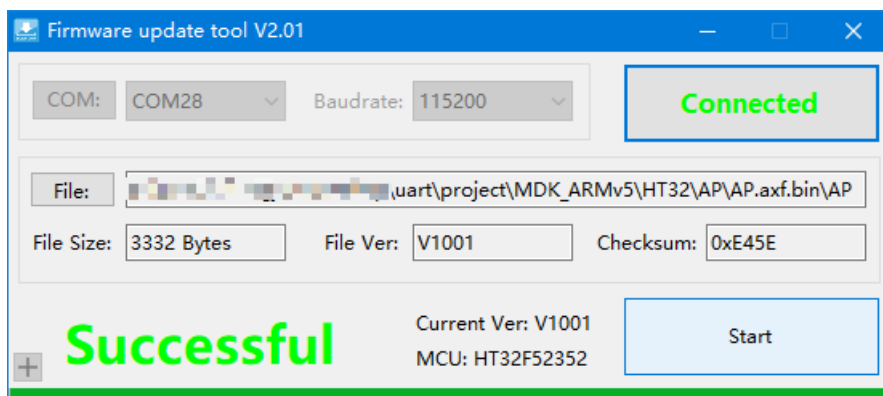
名称	修改日期	类型
AP.uvoptx	2019-12-25 14:32	UVOPTX 文件
AP.uvprojx	2019-12-25 14:32	UVPROJX 文件
ht32_op.s	2019-08-02 20:14	Assembler Source
HT32F5xxxx_01_DebugSupport.ini	2019-07-08 12:58	配置设置
iap.s	2019-12-25 14:32	Assembler Source
IAP.uvprojx	2019-12-25 14:32	UVPROJX 文件
linker.lin	2019-12-25 14:32	LIN 文件
startup_ht32f5xxxx_01.s	2019-12-25 14:32	Assembler Source
UART_IAP.lib	2018-08-31 14:27	Altium Library
UART_IAP.uvmpw	2019-12-25 14:32	UVMPW 文件



- 设置 AP 为 Active Project
- Batch Build IAP 和 AP 两个工程
- 下载到 MCU；建议下载之后稍微修改代码，例如点亮一个 LED，然后编译，但不要再通过 MDK 下载到 MCU。用于验证更新后程序功能有变化。
- 用串口线连接 MCU 到电脑
- 打开 HT32_UART_IAP_Tool.exe，选择串口号，波特率默认 115200，connect



- 成功连接后会显示当前 MCU 的型号和当前的固件版本号，然后可以选择更新的 bin 文件，bin 文件一般位于.\project\MDK_ARMv5\HT32\AP\AP.axf.bin 文件夹内。

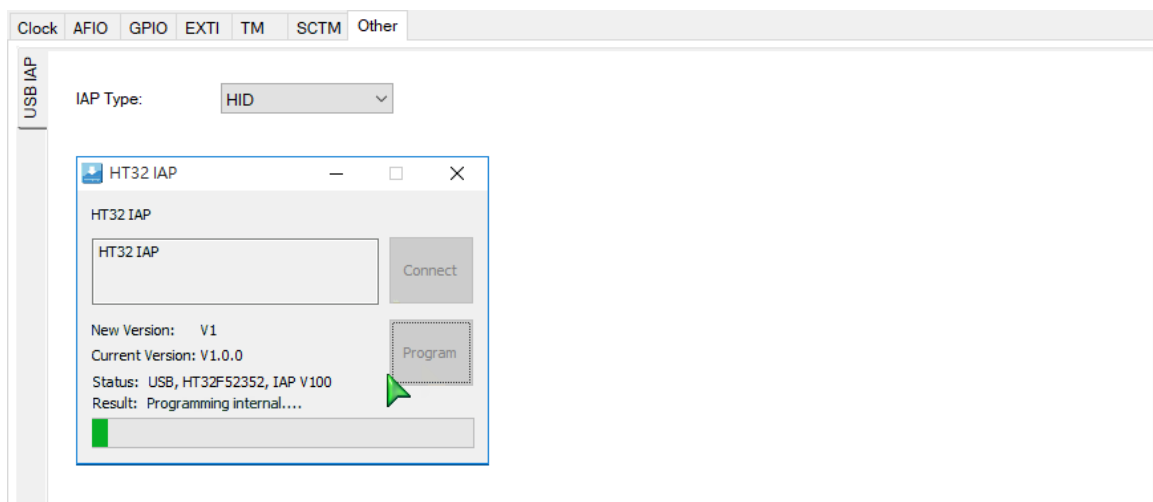


- 文件选择正确则会显示其版本号和 checksum，点击 start 开始下载更新，成功后会提示 successful。

20.4 USB IAP 设置：

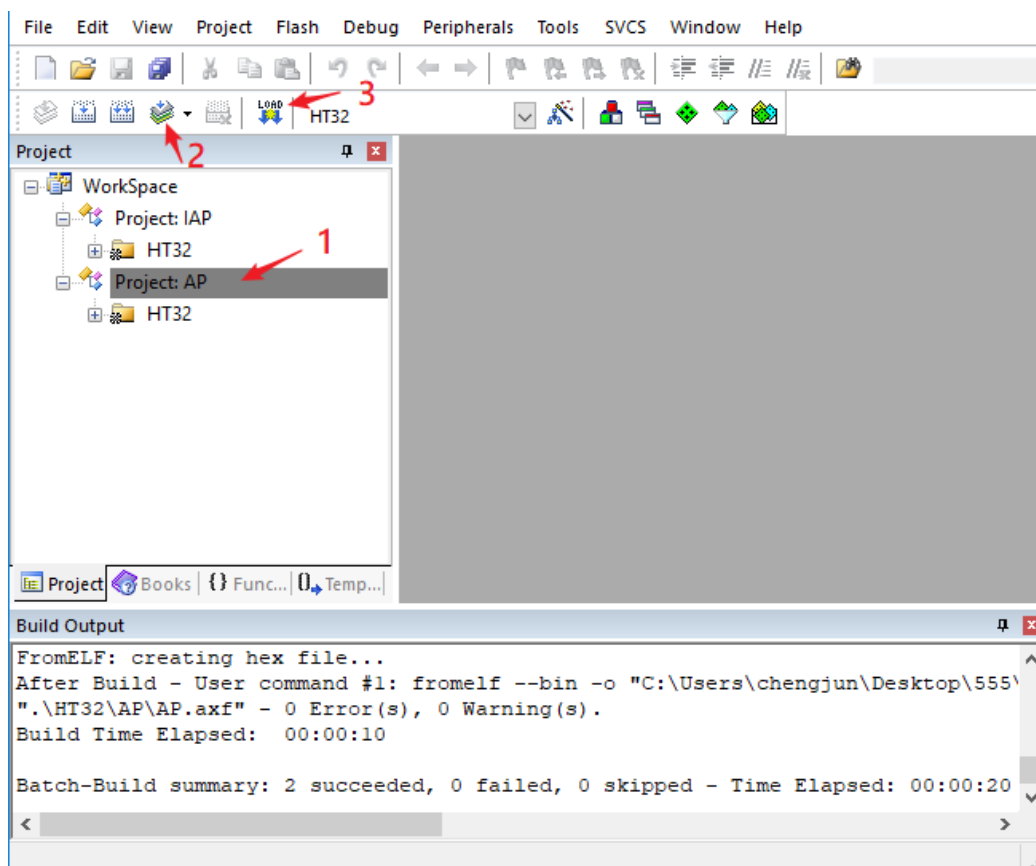
如果 MCU 具有 USB 功能，就可以通过菜单栏 Tools→Add USB IAP 激活此功能。USB IAP 分为两个类型，一个是 HID 类，一个是 mass storage 类，下面详细介绍。

20.4.1 USB HID IAP

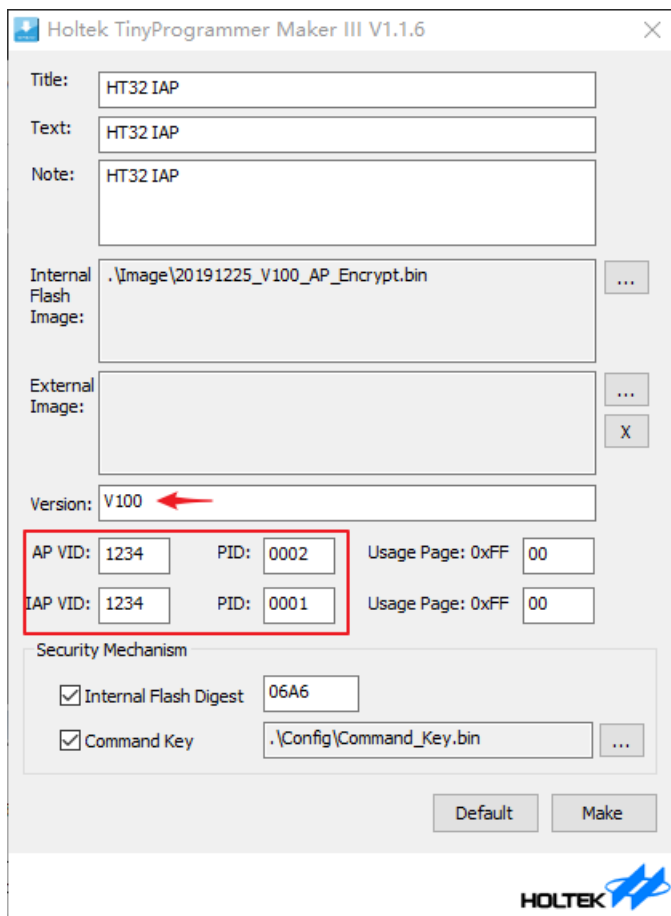


- 激活 USB IAP 功能后，在上图页面 IAP type 选择 HID；
- 如果有 GPIO, EXTI, TM, SPI, UART, I2C 等其它需要设置的，请一并设置好，按 build 流程输出工程。
- 生成的工程包含 IAP, AP 两个工程，点击 Multiproject 可以打开两个工程，如下图：

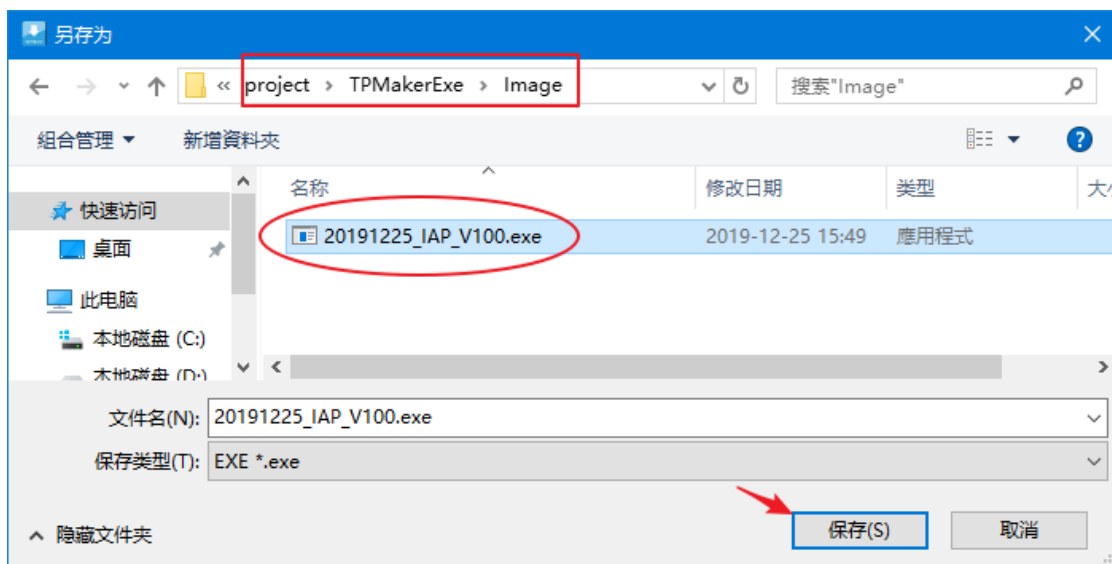
名称	修改日期	类型
_HT32_USBBufCheck.lib	2018-04-09 16:34	Altium Library
AP.uvoptx	2019-12-25 15:47	UVOPTX 文件
AP.uvprojx	2019-12-25 15:47	UVPROJX 文件
ht32_op.s	2019-08-02 20:14	Assembler Source
HT32F5xxxx_01_DebugSupport.ini	2019-07-08 12:58	配置设置
iap.s	2019-12-25 15:47	Assembler Source
IAP.uvprojx	2019-12-25 15:47	UVPROJX 文件
linker.lin	2019-12-25 15:47	LIN 文件
startup_ht32f5xxxx_01.s	2019-12-25 15:47	Assembler Source
USB_HID_IAP.uvmpw	2019-12-25 15:47	UVMPW 文件
version.s	2019-12-25 15:47	Assembler Source
version.txt	2019-12-25 15:47	文本文档



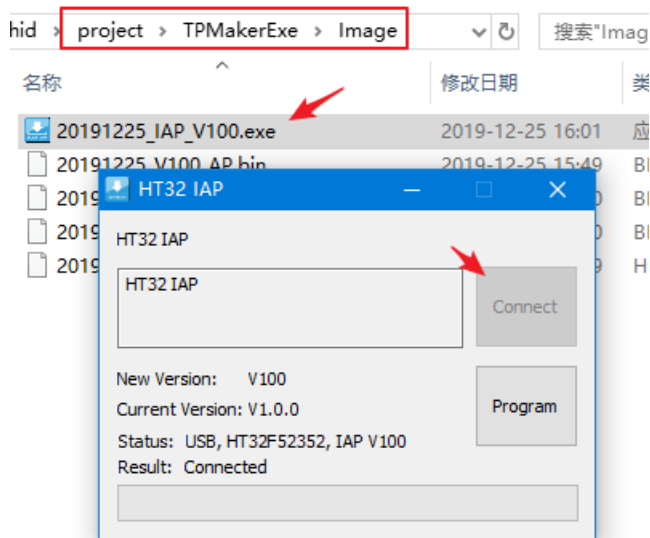
- 设置 AP 为 Active Project
- Batch Build IAP 和 AP 两个工程，工程编译结束时会弹出 TinyProgrammer Maker，输入版本号，点击 Make 按钮（VID/PID 的值，是根据固件的源代码，在.\project\TPMakerExe\Config\make_config.bat 中设定的，如果修改请与固件源代码对应位置一起修改）



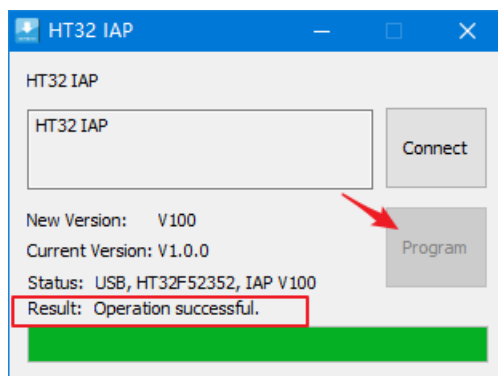
- 选择如下路径，指定已经按日期和版本号取好名字的文件，覆盖保存



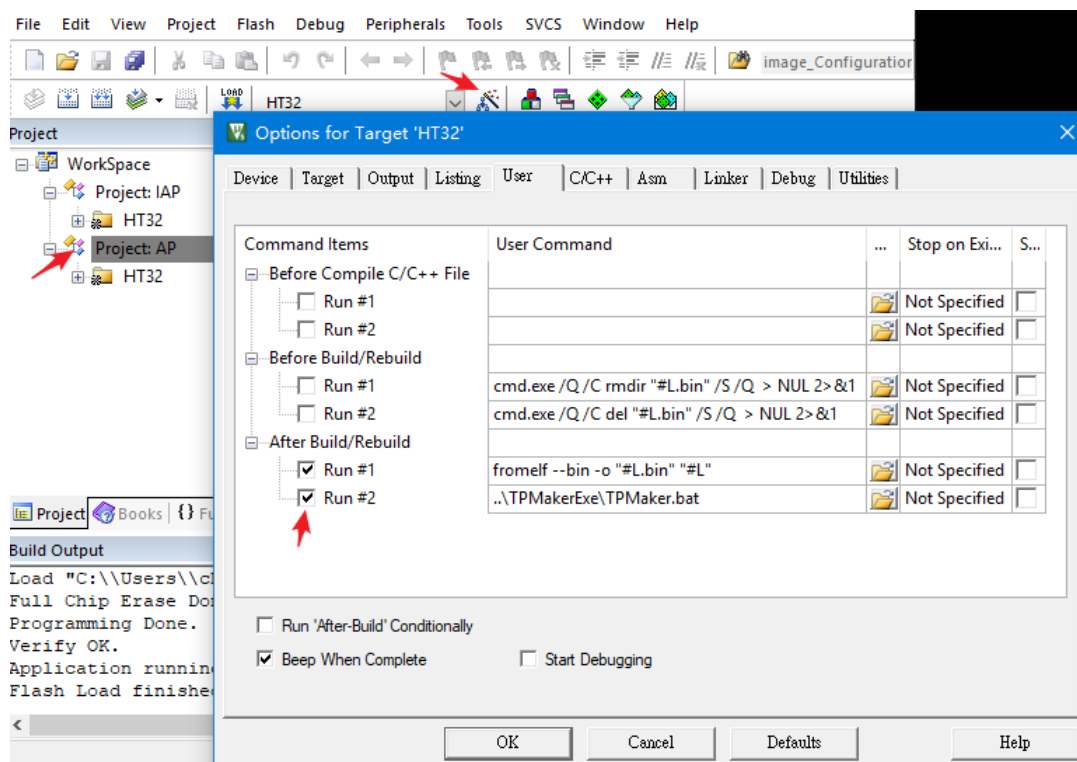
- 下载到 MCU；建议下载之后稍微修改代码，例如点亮一个 LED，然后编译，按上面的步骤重新制作一个 exe 文件，但不要再通过 MDK 下载到 MCU。用于验证更新后程序功能有变化。
- 用 USB 线连接 MCU 到电脑
- 打开刚才生成的 exe 文件，点击 connect，连接成功后会显示芯片的型号，和当前芯片内的固件版本号，以及本次更新的版本号，如下图。



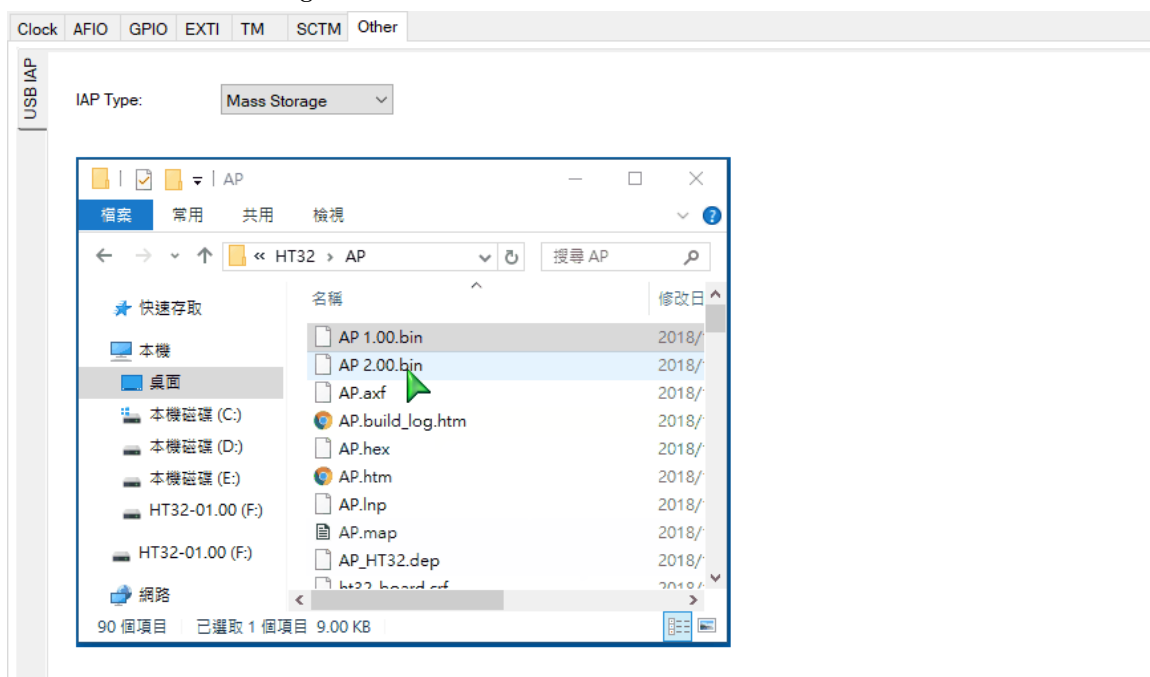
- 点击 Program 就开始更新了，更新完成后会立即执行新的固件



- 如果不希望每次编译 AP 工程都进行升级文件的制作流程，请于 AP 的 option→user 页面取消 Run #2 的选择，如下图：

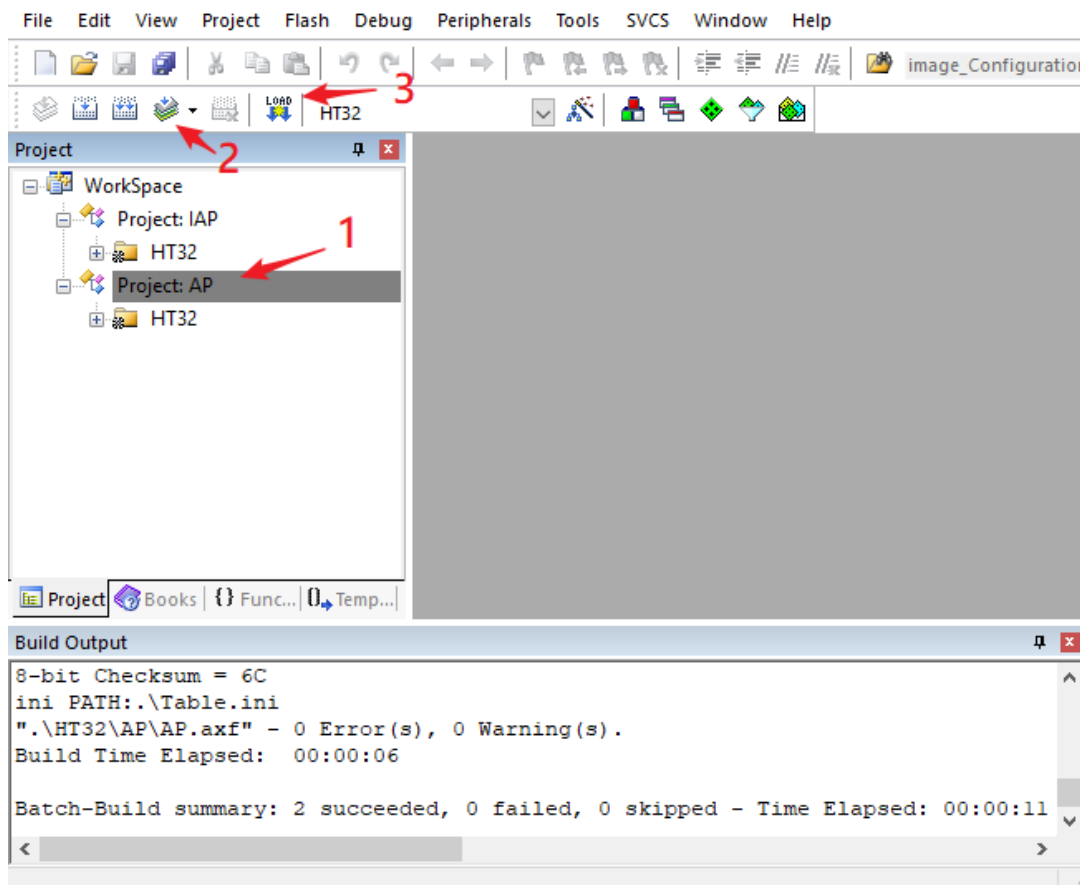


20.4.2 USB mass storage IAP

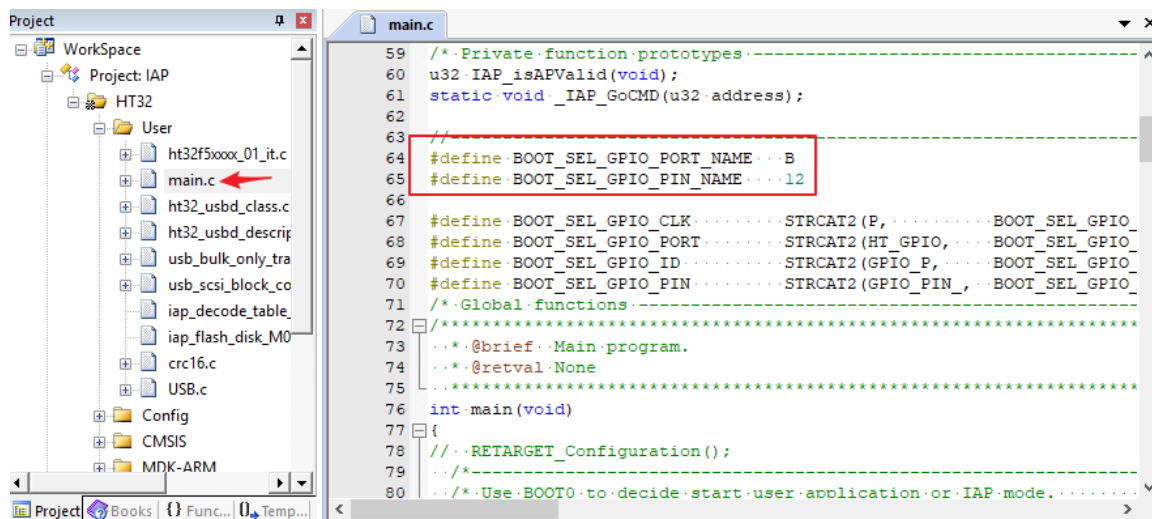


- 激活 USB IAP 功能后，在上图页面 IAP type 选择 Mass Storage；
- 如果有 GPIO, EXTI, TM, SPI, UART, I2C 等其它需要设置的，请一并设置好，按 build 流程输出工程。
- 生成的工程包含 IAP, AP 两个工程，点击 Multiproject 可以打开两个工程，如下图：

project > MDK_ARMv5		搜索"MDK_ARMv5"	
名称	修改日期	类型	大
AP.uvoptx	2019-12-25 16:20	UVOPTX 文件	
AP.uvprojx	2019-12-25 16:20	UVPROJX 文件	
hex_encode.exe	2018-10-05 19:30	应用程序	
ht32_op.s	2019-08-02 20:14	Assembler Source	
HT32F5xxxx_01_DebugSupport.ini	2019-07-08 12:58	配置设置	
iap.s	2019-12-25 16:20	Assembler Source	
IAP.uvprojx	2019-12-25 16:20	UVPROJX 文件	
iap_decode_table_M0_plus.lib	2018-10-16 17:56	Altium Library	
iap_flash_disk_M0_plus.lib	2018-10-16 17:56	Altium Library	
linker.lin	2019-12-25 16:20	LIN 文件	
startup_ht32f5xxxx_01.s	2019-12-25 16:20	Assembler Source	
Table.ini	2019-12-25 16:20	配置设置	
USB_MassStorage_IAP.uvmpw	2019-12-25 16:20	UVMPW 文件	

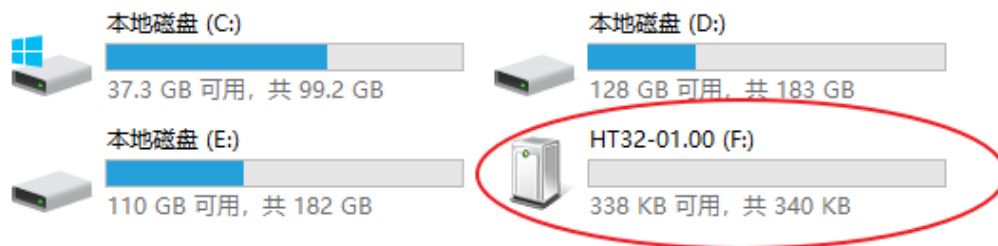


- 设置 AP 为 Active Project
- Batch Build IAP 和 AP 两个工程
- 下载到 MCU；建议下载之后稍微修改代码，例如修改磁盘名称，增加一个 LED 功能，然后编译但不要再通过 MDK 下载到 MCU。用于验证更新后程序功能的变化。
- 用 USB 线连接 MCU 到电脑
- IAP 工程里的默认设定是 PB12 拉低则程序停留在 IAP 里，也可以修改 AP 程序，通过某种触发条件，例如按某个按键，进入 IAP。这个部分太灵活，此处不深入讨论。按照 PB12 拉低复位一次使程序进入 IAP。

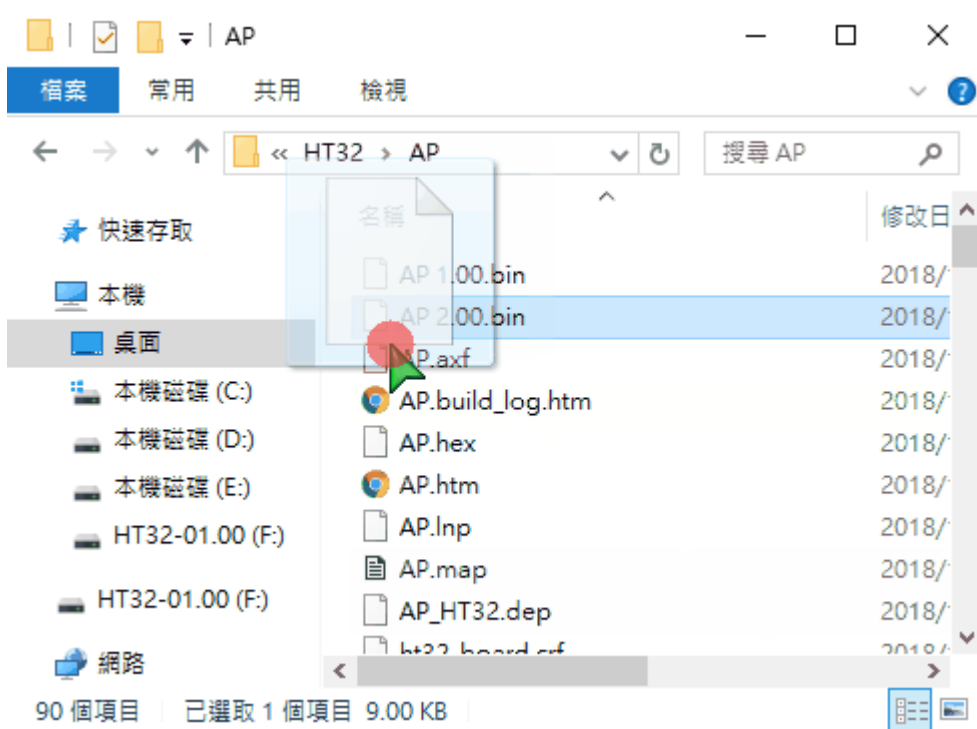


- 进入 IAP 程序之后，电脑会弹出一个可移动磁盘，如下图。

▼ 设备和驱动器 (4)



- 把需要更新的 bin 文件（位于.\project\MDK_ARMv5\HT32\AP\AP.bin），拖到弹出的可移动磁盘里，更新就完成了。



- 保持 PB12 拉高，手动复位一次，开始执行新的固件。
- 文件.\project\MDK_ARMv5\ Table. ini 是加密表，若不希望他人破解用于更新的 bin 文件，请自行修改 Table. ini 的数据，注意 IAP 的 main. c 里 Table1[139] 需要与此文件的数据一致。
- 这种更新方式可用在 Windows, MAC, Linux 各种操作系统上。

21. 版本历史

Revision	Date	Description
1.00	2018-05-07	The first Public release version
1.01	2018-06-28	增加 M0+ 的几个新型号的支持; 增加 M3 的支持; 修正一些 bug;
1.02	2018-11-20	增加了 UART IAP, USB IAP 的功能; 修正了很多小 bug; 可能会增加 bug T_T。
1.03	2018-12-05	修正 USART TOUT 中断未清对应标志 bug;
1.04	2019-1-17	修改 50xxx 系列的封装; 修正 50xxx 系列选 HSI 时的 bug; 加速程序运行速度;
1.05	2019-7-03	增加 PDMA 功能; 增加 HT32F0006, 增加 DAC, QSPI 功能; 修正一些 bug; 更新 FWlib 版本;
1.06	2019-8-22	增加 LCD_TFT 平台功能; 增加 USART 同步模式; 修改一些细节;
1.07	2019-12-20	增加几个 MCU 新型号; 更新了 FWlib, 更新 50xxx 系列的封装; 优化了一些细节问题;
1.08	2020-03-31	修正 12364, 57xxx 的配置文件错误; 增加可选输出 IAR 工程; 修正创建 SPI 文件的 bug; 修正重开保存的文件会少开某个页面的 bug;
1.09	2020-05-06	修正 12364, 502xx 的配置文件错误; 修正 UART IAP 使能 USB 功能后生成文件 bug;

Holtek Semiconductor Inc. (Headquarters)

No.3, Creation Rd. II, Science Park, Hsinchu, Taiwan

Tel: 886-3-563-1999

Fax: 886-3-563-1189

<http://www.holtek.com.tw>

Holtek Semiconductor Inc. (Taipei Sales Office)

4F-2, No. 3-2, YuanQu St., Nankang Software Park, Taipei 115, Taiwan

Tel: 886-2-2655-7070

Fax: 886-2-2655-7373

Fax: 886-2-2655-7383 (International sales hotline)

Holtek Semiconductor (China) Inc.

Building No.10, Xinzhu Court, (No.1 Headquarters), 4 Cuizhu Road, Songshan Lake, Dongguan, China 523808

Tel: 86-769-2626-1300

Fax: 86-769-2626-1311

Holtek Semiconductor (USA), Inc. (North America Sales Office)

46729 Fremont Blvd., Fremont, CA 94538, USA

Tel: 1-510-252-9880

Fax: 1-510-252-9885

<http://www.holtek.com>

Copyright © 2013 by HOLTEK SEMICONDUCTOR INC.

The information appearing in this Data Sheet is believed to be accurate at the time of publication. However, Holtek assumes no responsibility arising from the use of the specifications described. The applications mentioned herein are used solely for the purpose of illustration and Holtek makes no warranty or representation that such applications will be suitable without further modification, nor recommends the use of its products for application that may present a risk to human life due to malfunction or otherwise. Holtek's products are not authorized for use as critical components in life support devices or systems. Holtek reserves the right to alter its products without prior notification. For the most up-to-date information, please visit our web site at <http://www.holtek.com>