

前言

本手册用于介绍如何使用雅特力科技提供的时钟配置工具来进行对AT32 MCU时钟路径及参数的配置，并生成相应的时钟配置流程代码。

支持型号列表：

支持型号	AT32F403xx
	AT32F403Axx
	AT32F407xx
	AT32F413xx
	AT32F415xx
	AT32F421xx

目录

1	简介	5
1.1	环境要求	5
1.2	安装	5
2	功能介绍	6
2.1	菜单栏	7
2.2	新建配置项目	7
2.3	配置界面的使用	8
2.4	生成代码	9
3	源码使用	10
3.1	使用准备	10
3.2	Keil 工程	10
3.3	IAR 工程	12
3.4	外部时钟源(HSE)修改方法	15
3.5	注意事项	16
4	版本历史	17

表目录

表 1. 文档版本历史	17
-------------------	----

图目录

图 1. 启动界面	6
图 2. 配置界面	6
图 3. 菜单栏界面.....	7
图 4. MCU 选择界面	8
图 5. 配置界面框架	8
图 6. 时钟配置框.....	9
图 7. Keil 增加 PLLconfig 管理组	10
图 8. Keil 添加源码文件	11
图 9. Keil 增加 Include 路径.....	12
图 10. IAR 增加管理组	13
图 11. IAR 设置管理组名	14
图 12. IAR 添加源码文件	14
图 13. IAR 增加 Include 路径.....	15

1 简介

AT32时钟配置工具是雅特力科技为方便对AT32系列MCU进行时钟配置而开发的一个图形化配置工具，其主旨是让用户清晰了解时钟路径和配置出期望的时钟频率。

1.1 环境要求

■ 软件要求

需要Windows XP、Windows7及以上操作系统支持。

需要.net framework4.0的支持。对于部分未安装.net framework4.0的操作系统，需要安装.net framework4.0。

■ 硬件要求

PC/AT兼容机，Penium或更高CPU。

XVGA(1024*768)颜色显示器。

至少512MB RAM。

至少20MB磁盘空间。

1.2 安装

■ 软件安装

本软件不需要安装，只需直接运行可执行程序AT32 Clock Configuration.exe。

2 功能介绍

本章节将介绍此工具的基本操作，其主要的启动界面和配置界面如下所示

图 1. 启动界面

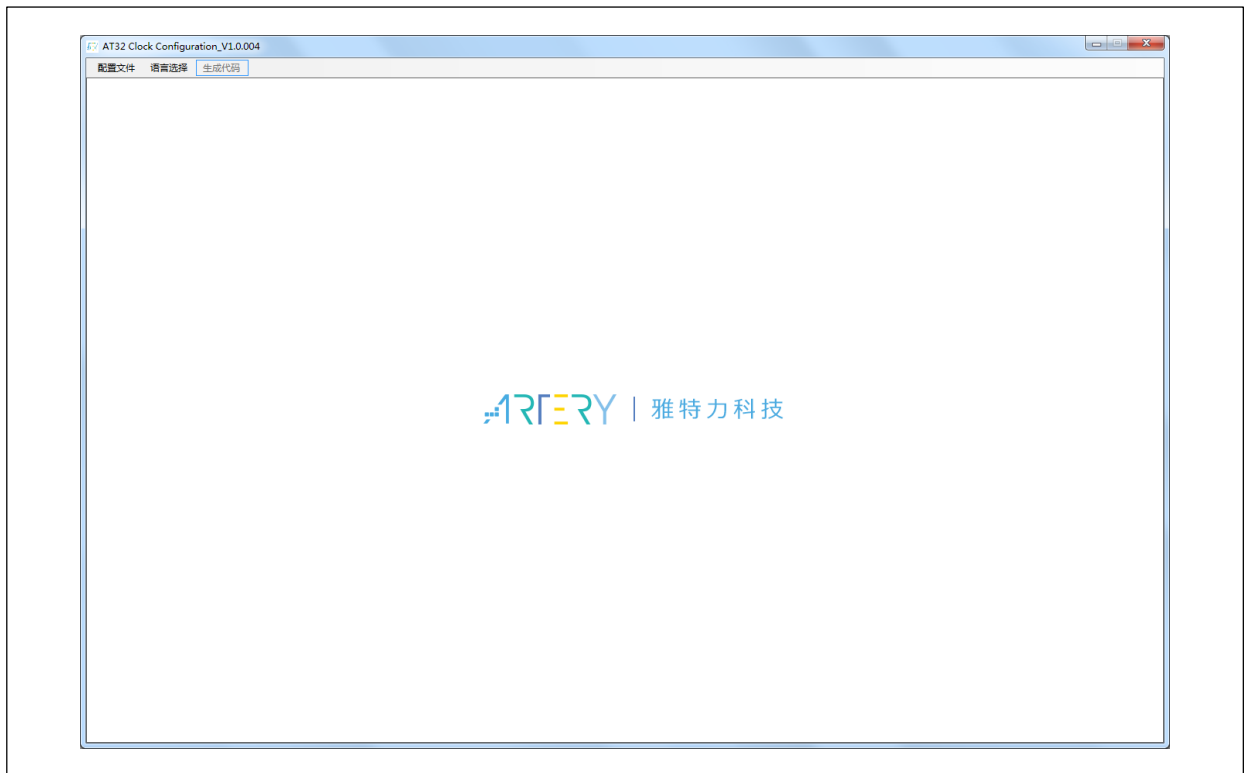
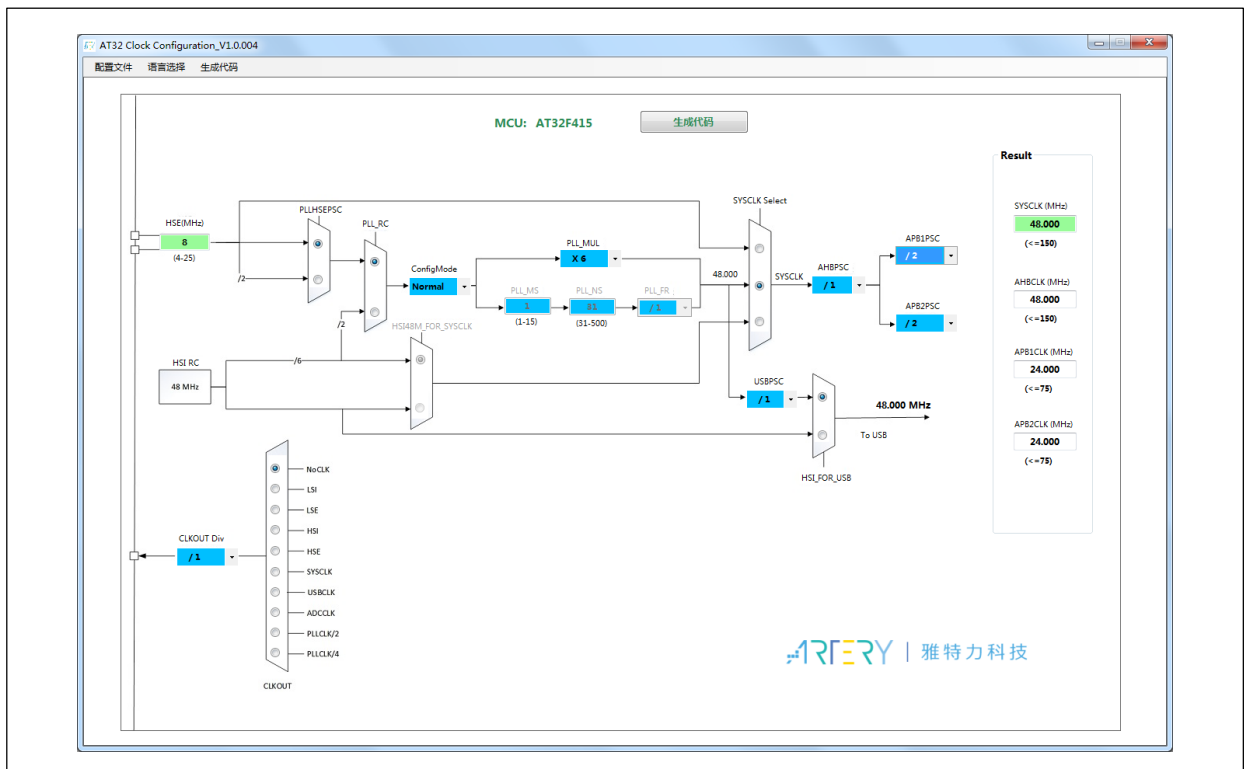


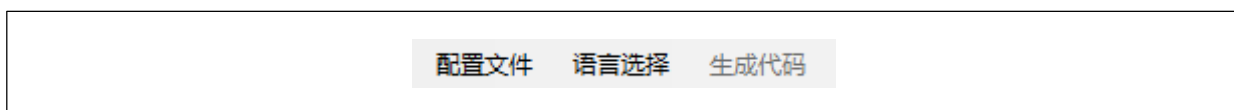
图 2. 配置界面



2.1 菜单栏

菜单栏内容如图所示：

图 3. 菜单栏界面



■ “配置文件” 菜单：

- 新建： 新建时钟配置项目
- 打开： 打开已存在的配置项目
- 关闭： 关闭已打开的配置项目
- 保存： 保存已打开的配置项目
- 另存为： 另存为已打开的配置项目
- 退出： 退出上位机配置软件

■ “语言选择” 菜单：

- English： 选择 English 作为显示语言
- 简体中文： 选择简体中文作为显示语言

■ “生成代码” 菜单：

当在对应型号的操作配置界面将所期望的时钟路径和时钟频率配置完成之后，可点击“生成代码”菜单来选择源码文件的存储路径并生成相应的时钟配置源码文件（PLLconfig.c 和 PLLconfig.h）。

2.2 新建配置项目

双击打开时钟配置工具，可看到如图 1 所示的启动界面，可点击“配置文件”菜单-->“新建”，进行配置项目的新建，在新建配置项目的过程中需要对芯片的系列所属进行选择，操作方法如下图所示

图 4. MCU 选择界面



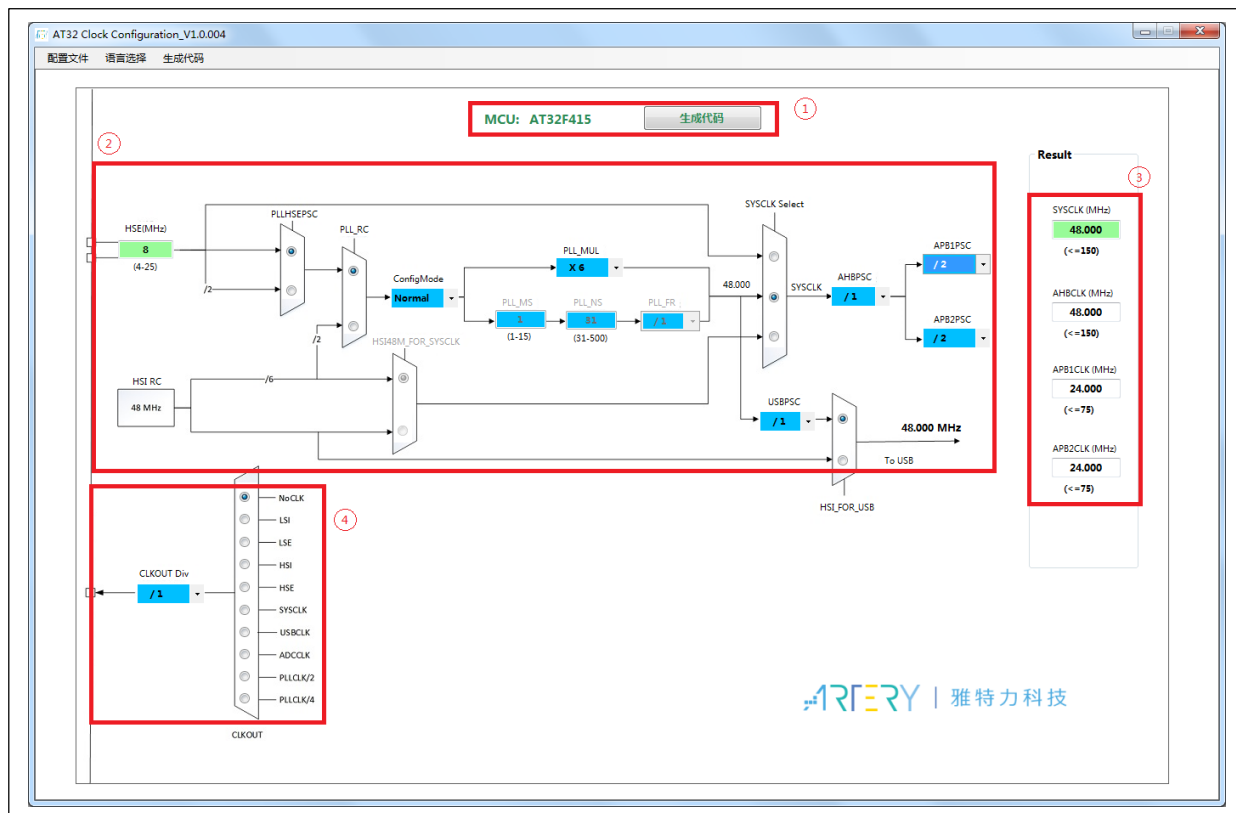
MCU 系列的选择，可点击下来框来进行选择，目前仅支持 AT32F403、AT32F403A、AT32F407、AT32F413、AT32F415 和 AT32F421 系列（实际的支持情况视版本而定），当选择好 MCU 后点击“确定”可进入到与图 2 类似的配置界面。

2.3 配置界面的使用

配置界面主要用来进行时钟路径及参数的配置，以下的介绍将以 AT32F415 系列作为示例来展开进行，其余系列的配置方法与此类似。

整个配置界面主要可以分为四个大块，如下图所示

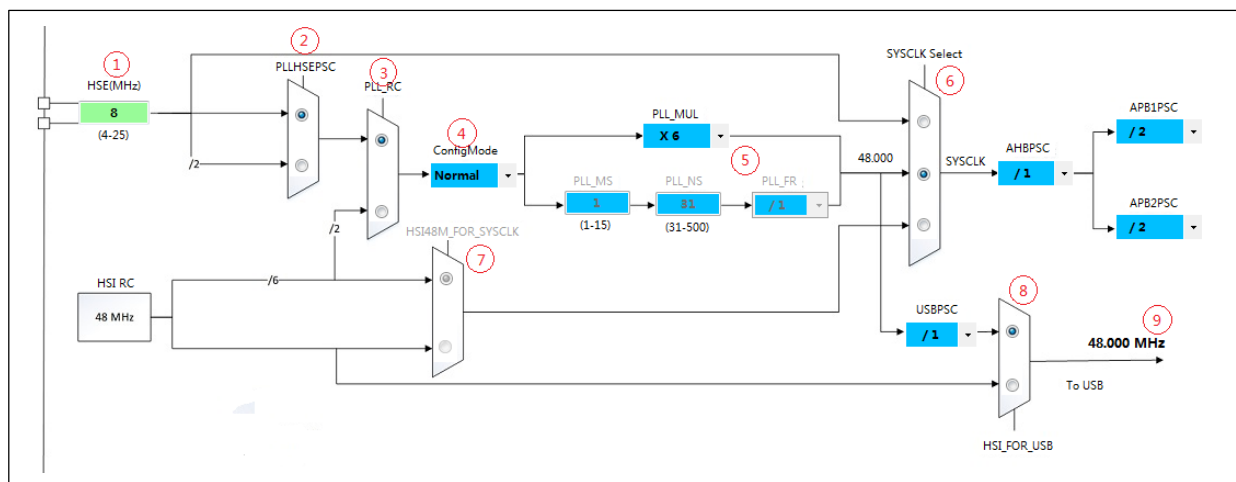
图 5. 配置界面框架



- ① 标题部分：用于展示当前配置项目所选择的 MCU 系列，和“生成代码”按钮。
- ② 配置部分：用于对时钟路径和时钟参数进行选择 and 配置，以达到期望的应用需求。
- ③ 结果部分：用于显示配置部分所配置出的时钟频率结果，在 **SYSCLK** 栏也可在选中 PLL 为系统时钟时作为输入框，可输入期望的系统时钟频率来反向自动配置出倍频参数。
- ④ 输出部分：用于时钟输出（**CLKOUT**）的配置。

接下来就着重介绍一下配置部分的使用。配置部分的流程界面是对应着 MCU 时钟树来进行的，各系列 MCU 的此部分可能存在着差异，但使用方式大同小异。时钟路径的配置可按流程对各开关进行点选来进行选择，配置部分如下图所示，将逐个流程点的功能及其注意事项进行介绍。

图 6. 时钟配置框



- ① **HSE**：此为输入框，8 MHz 为所采用外部时钟源的默认频率，用户可根据实际使用的外部时钟频率进行修改。（注：此 8 MHz 被修改为其他频率值时，对应的 BSP 中 Libraries\CMSIS\CM4\DeviceSupport\at32f4xx.h 文件内的 HSE_VALUE 宏定义也应该一致修改）。

- ② PLLHSEPSC: 点选框, 当 HSE 作为 PLL 时钟源时, 可配置输入频率为 HSE 2 除频或 HSE 不除频。
- ③ PLLRC: 点选框, 可配置 PLL 时钟源为 HSE 或 HSI。
- ④ ConfigMode: 下拉框, 可选择常规模式 (Normal) 或灵活模式 (Flexible), 默认采用常规模式 (注: 此功能目前主要针对于 AT32F415 和 AT32F421 系列使用, 对于灵活模式配置功能的详细说明请参考 Reference Maunul 的 PLL 章节)。
- ⑤ 倍频系数: 此处倍频系数有两组, 分别对应着配置模式中的常规模式 (Normal) 和灵活模式 (Flexible)。常规模式使用 PLL_MUL 参数进行倍频, 计算公式为: PLL 时钟 = PLL 输入时钟 * PLL_MUL, 灵活模式使用 PLL_MS、PLL_NS 和 PLL_FR 三个参数来共同决定倍频系数, 计算公式为: PLL 时钟 = (PLL 输入时钟 * PLL_NS) / (PLL_MS * PLL_FR), 各项参数的限制条件可以参考 Reference Maunul 的 PLL 章节。为了用户的使用方便, 本工具在选定 PLL 输入时钟源和配置模式后, 在结果部分的 SYSCLK 框中输入目标时钟并按下 “Enter” 键, 会自动计算一组倍频参数以满足用户期望或相近的时钟频率。
- ⑥ SYSCLK select: 点选框, 可配置 HSE、PLL 或 HSI 作为系统时钟。
- ⑦ HSI48M_FOR_SYSCLK: 点选框, 当 SYSCLK select 选择 HSI 作为系统时钟时, 可配置 8 MHz 或 48 MHz HSI RC 到系统时钟 (注: 当选择 48 MHz HSI 到系统时钟后, CLKOUT 输出 HSI 时的频率也为 48 MHz)。
- ⑧ HSI_FOR_USB: 点选框, 可配置 USB 时钟由 PLL 时钟或 HSI 48 MHz 提供, 在选中 HSI 48 MHz 到 USB 时, 会默认开启 ACC 时钟校准。因 USB 时钟需要的是固定 48 MHz, 所以在 USB PSC 可分频参数为前提下, PLL 倍频出来的频率可能不满足 USB 48 MHz 的时钟需求。
- ⑨ USB 时钟频率的显示。此显示栏会实时计算 USB 时钟的频率并显示, 如果配置出来的 USB 时钟不等于 48 MHz 时, 显示出来的 USB 时钟频率会标注为红色, 而实际应用中并没有用到 USB 时可对此忽略。(注: 此部分只针对 USB 时钟频率的配置, USB 外设时钟使能需自行额外打开)

2.4 生成代码

当时钟配置完成后, 可点击生成代码, 然后选择代码生成的目录并确认, 最后会在所选目录下生成两份文件 (PLLconfig.c 和 PLLconfig.h)。这两份源码文件可包含到项目工程内进行使用。

3 源码使用

本章节将对时钟配置工具所生成的配置源码文件如何使用进行介绍。源码的使用可能会因为开发工具的不同有细微差异，接下来会分别使用常用的开发工具 Keil 和 IAR 基于 AT32 BSP 的 Templates 工程来进行操作和介绍。

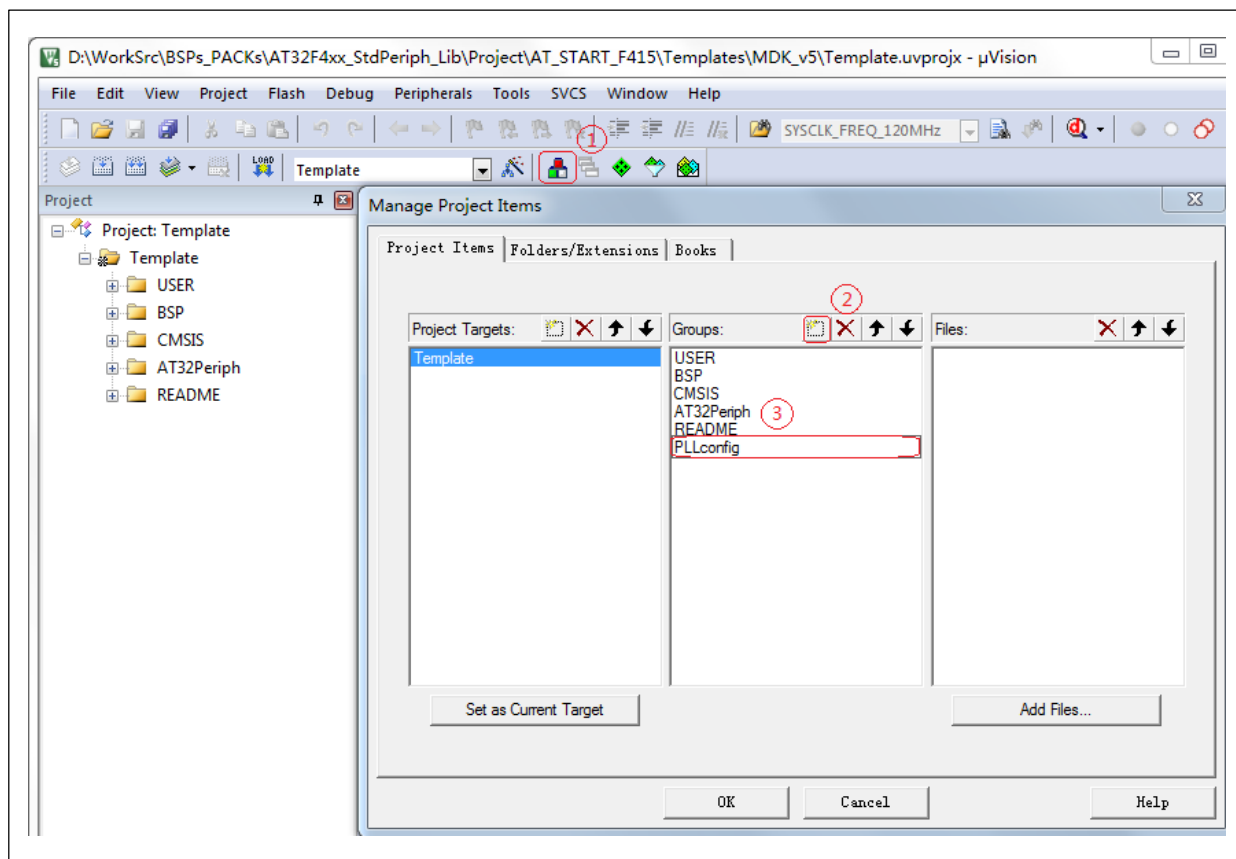
3.1 使用准备

在 AT32 BSP 包的 AT32F4xx_StdPeriph_Lib_V1.x.x\Project\AT_START_F415\Templates 目录下新建文件夹 PLLconfig，并且时钟配置工具生成代码到此文件夹下。

3.2 Keil 工程

打开 AT32F4xx_StdPeriph_Lib_V1.x.x\Project\AT_START_F415\Templates\MDK_v5 下 Template 工程，首先在工程文件管理项中新增 PLLconfig 管理组，步骤如下图

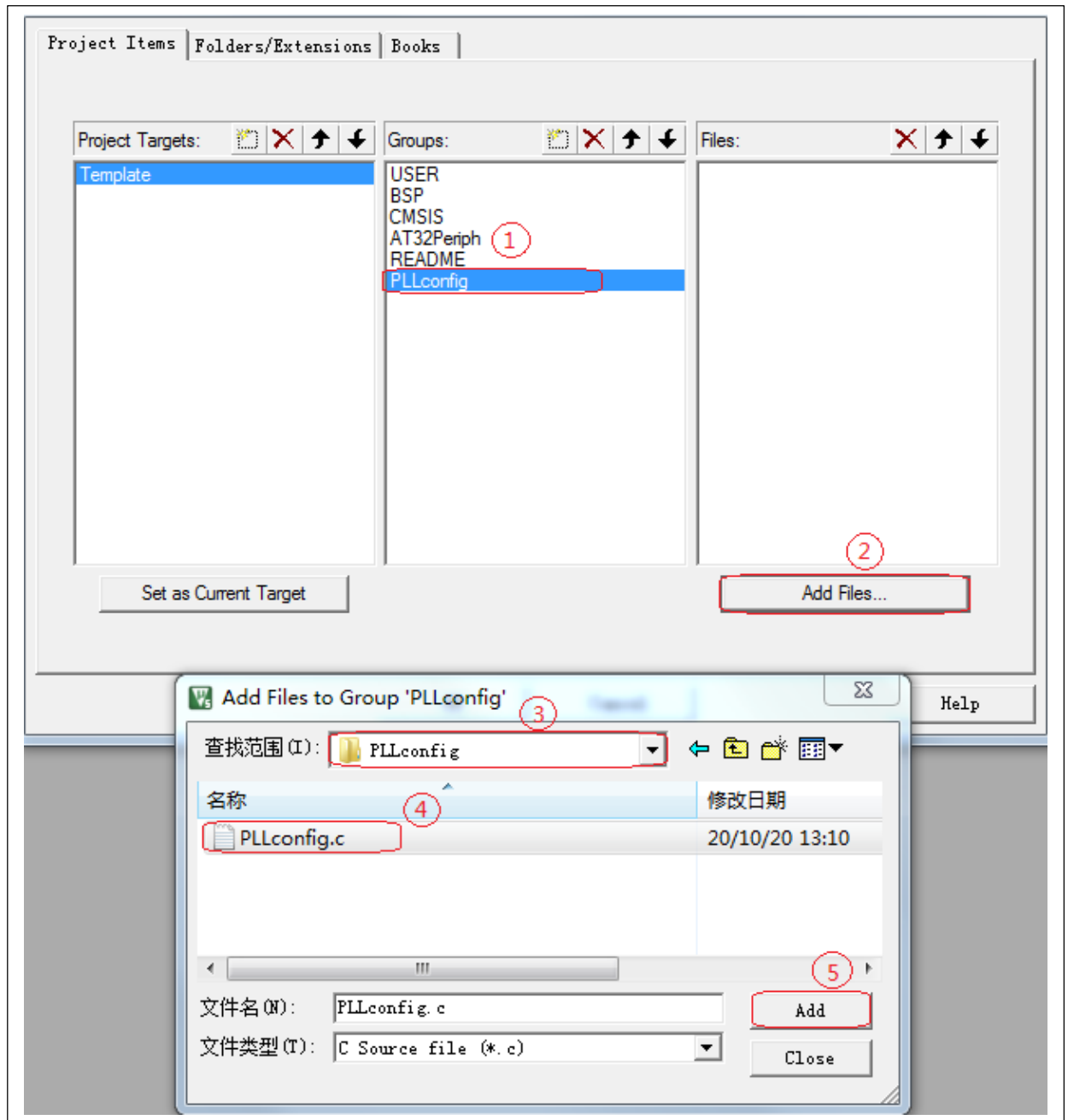
图 7. Keil 增加 PLLconfig 管理组



- ① 点击“File Extensions...”。
- ② 点击“New”。
- ③ 在新建的框内输入管理组名“PLLconfig”。

在管理组新建完成后，需将新生成的源码文件导入到建立好的文件栏中，步骤如下图

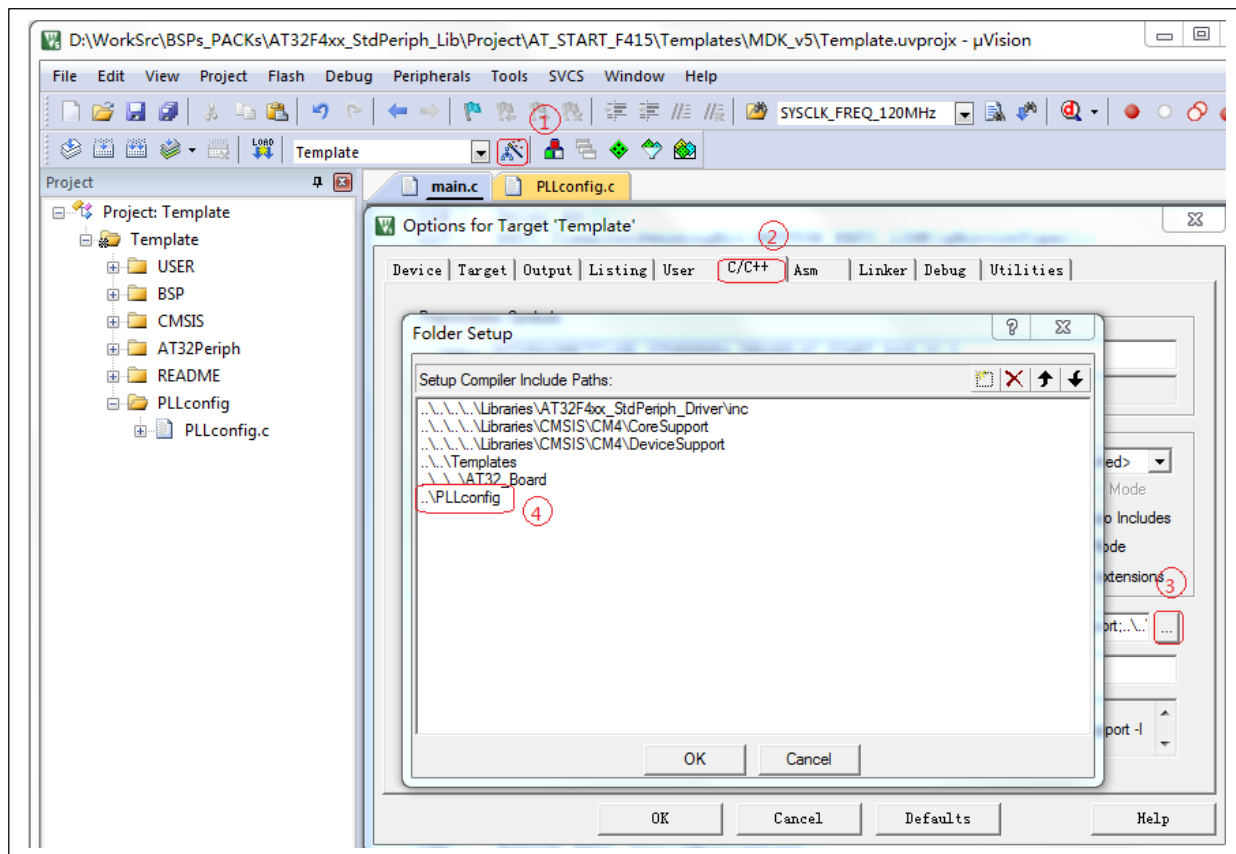
图 8. Keil 添加源码文件



- ① 点击建立好的管理组名 PLLconfig，被点击后背景色为蓝色。
- ② 点击“Add Files...”按钮。
- ③ 在查找范围栏浏览到新建的 PLLconfig 文件夹。
- ④ 点击选中文件夹下由时钟配置工具所生成的 PLLconfig.c 源码文件。
- ⑤ 点击“Add”按钮，最后点击“OK”按钮退出。

在源码文件导入到工程后，需要头文件的包含路径也加入到工程内，步骤如下图

图 9. Keil 增加 Include 路径



- ① 点击魔术棒“Options for Target...”。
- ② 选中“C/C++”选项卡。
- ③ 点击复选按钮。
- ④ 在空白栏双击并输入“..\PLLconfig”（注：此路径按实际情况的工程文件相对路径而定）。最后点击“OK”按钮并退出。

以上步骤完成之后，我们可以在工程管理组看见新增的 PLLconfig，和子集包含的 PLLconfig.c 文件。以上表示文件导入成功。

文件导入成功以后，就可以在应用代码中调用时钟配置函数。在调用之前应注释掉原工程中 system_at32f4xx.c 文件默认开启的系统时钟配置的宏定义（详情请参考 3.4 节外部时钟源(HSE)修改方法），以下示例在 main.c 中进行时钟配置函数的调用。如下：

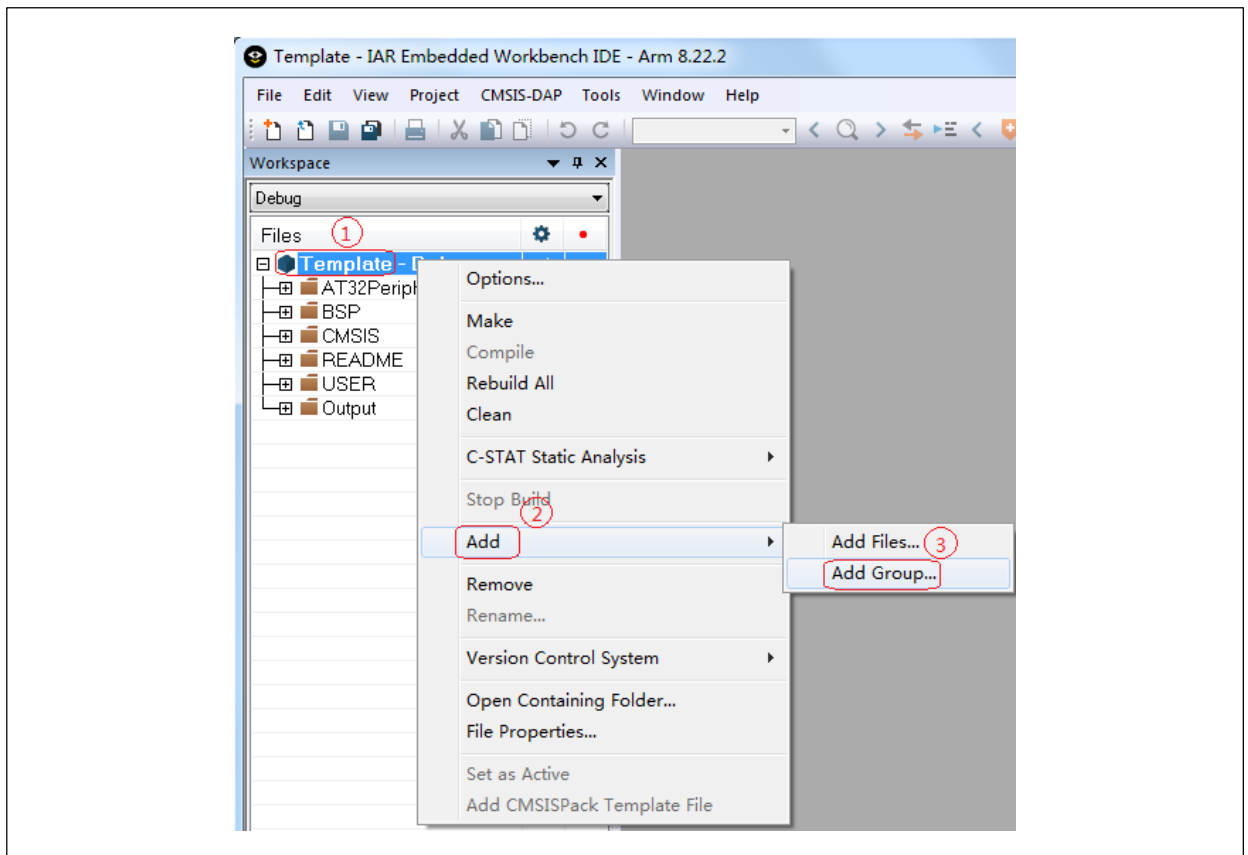
```
...  
#include "PLLconfig.h"  
...  
void main (void)  
{  
    ...  
    NewPLLconfig();  
    SystemCoreClockUpdate();  
    ...  
}
```

3.3 IAR 工程

IAR 工程下的操作步骤与 Keil 的配置一致，只是方法不一样而已。打开 AT32F4xx_StdPeriph_Lib_V

1.x.x\Project\AT_START_F415\Templates\IAR_v8.2 下 Template 工程，首先也是在管理项中新增 PLLconfig 管理组，步骤如下图

图 10. IAR 增加管理组



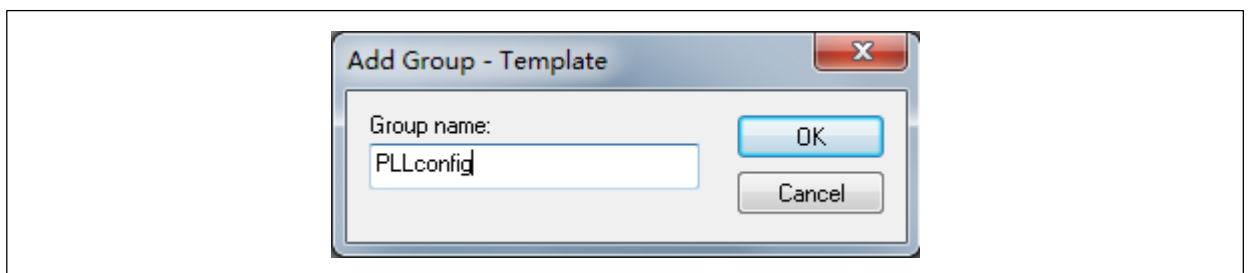
① 鼠标右击工程名“Template”。

② 选择“Add”。

③ 选择“Add Group...”并点击。

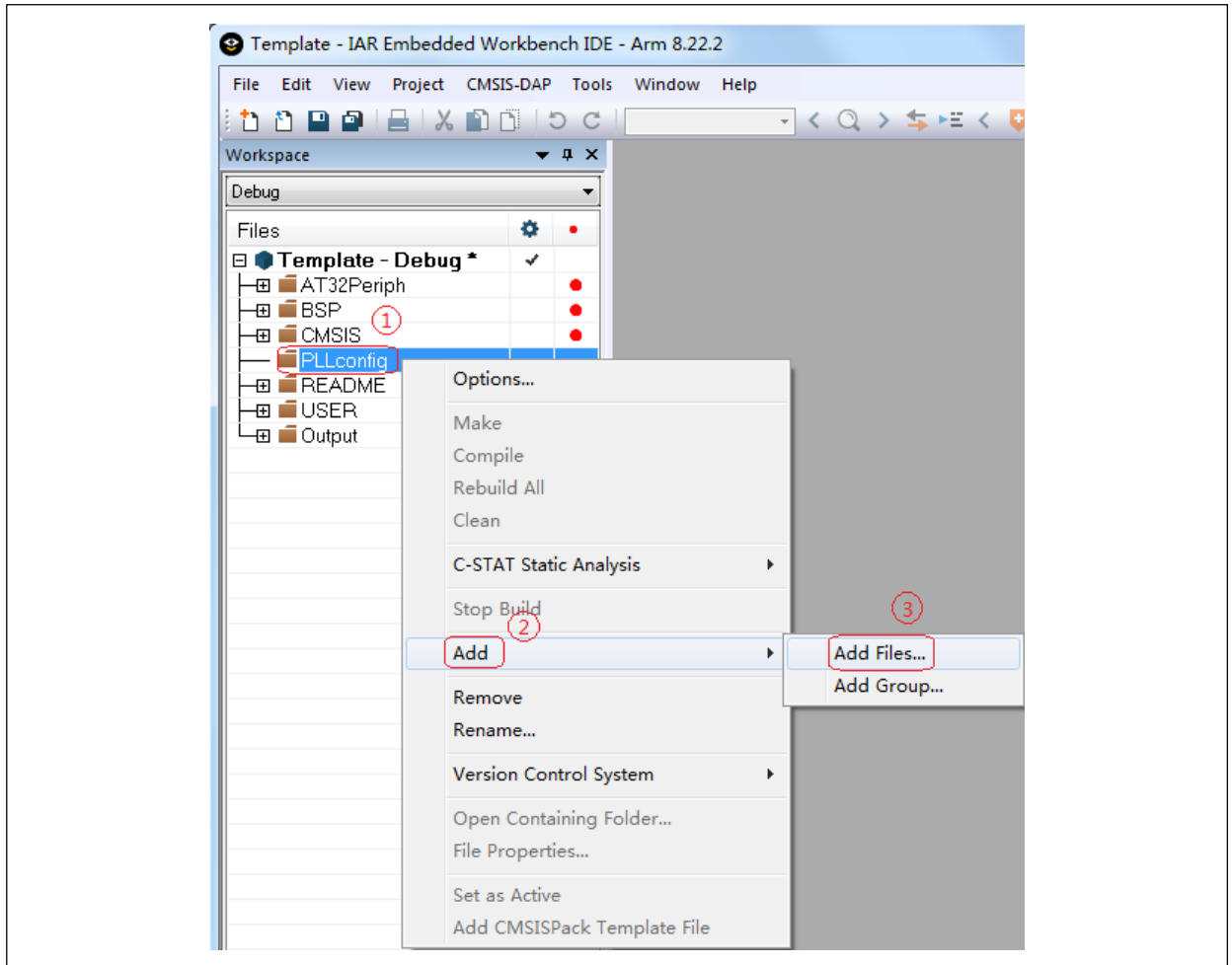
在弹出的对话框中输入管理组名“PLLconfig”，并点击“OK”。

图 11. IAR 设置管理组名



在新增的管理组中添加源码文件，步骤如下图

图 12. IAR 添加源码文件



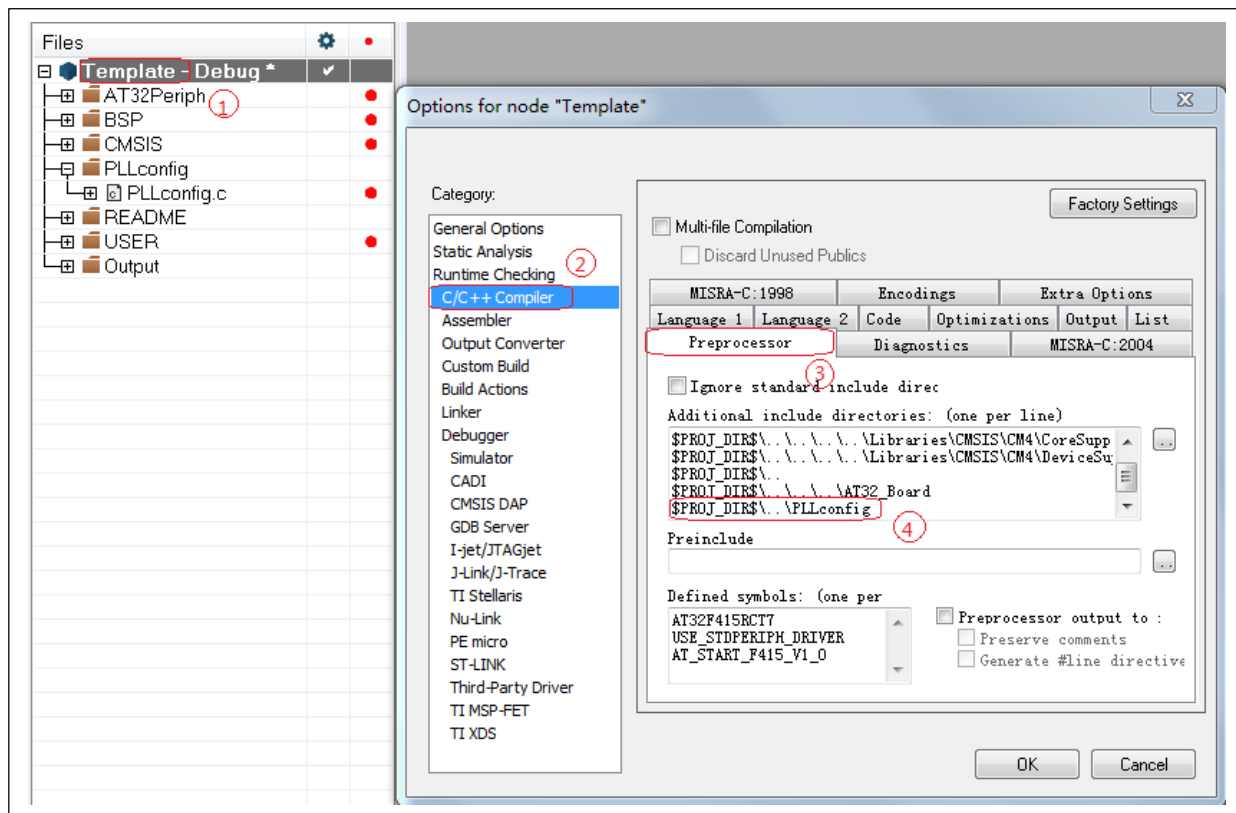
① 右击新增的管理组“PLLconfig”。

② 选择“Add”。

③ 选择“Add Files...”。在弹出框中选择到生成的 PLLconfig.c 文件并点击打开。

文件成功添加后可在工程管理栏看见新增的管理组“PLLconfig”和添加的源码文件“PLLconfig.c”。接下来添加头文件路径，IAR 中头文件路径加入工程的步骤如下图

图 13. IAR 增加 Include 路径



- ① 右击工程名并选择“Options...”。
- ② 选择到“C/C++Compiler”选项栏。
- ③ 点选“Preprocessor”选项卡。
- ④ 在空白栏双击并输入“\$PROJ_DIR\$\\..\\PLLconfig”，（注：此路径按实际情况的工程文件相对路径而定）。最后点击“OK”按钮并退出。

文件导入成功后，其函数调用方法与 Keil 中介绍的一样。

3.4 外部时钟源(HSE)修改方法

因本文档所示例的 Demo 和配置工具都默认采用的 8 MHz 外部时钟频率，当实际硬件使用的外部时钟源是非 8 MHz 频率时需注意修改以下几点。

- 1、在时钟配置工具中配置界面的 HSE 输入框内填入外部时钟源实际频率值并按“Enter”键确认。
- 2、注释掉原工程中 system_at32f4xx.c 文件默认开启的系统时钟配置的宏定义，将其修改为 HSI 作为系统时钟即可（以 AT32F415 Demo 示例就需要关闭 SYSCLK_FREQ_144MHz 宏定义）。

```
//#define SYSCLK_FREQ_144MHz      144000000 //需注意屏蔽
```

- 3、一定注意修改工程中使用的 Libraries\CMSIS\CM4\DeviceSupport\at32f4xx.h 文件内的 HSE_VALUE 宏定义值，需与时钟配置工具里 HSE 输入框内的频率值一致（如外部时钟源采用 12.288 MHz 晶振，需按如下修改）。

```
#define HSE_VALUE                ((uint32_t)12288000)
```

- 4、为保证 SysTick 等有用到 SystemCoreClock 变量的外设时钟初始化正常，最好在 NewPLLconfig() 函数调用后，再调用 SystemCoreClockUpdate() 函数来更新一下 SystemCoreClock 变量值。

```
NewPLLconfig();
SystemCoreClockUpdate(); //更新 SystemCoreClock 变量值
```

3.5 注意事项

在使用本时钟配置工具时需注意：

1. 此工具生成的时钟配置源码文件需结合雅特力科技提供的 **BSP** 进行使用。
2. 不同系列所生成的时钟配置源码文件不能型号混用，只能在相对应的工程项目中进行调用。
3. 配置工具中各输入框参数修改后，请以“Enter”键结束。

4 版本历史

表 1. 文档版本历史

日期	版本	变更
2020.10.20	1.0.0	最初版本

重要通知 - 请仔细阅读

买方自行负责对本文所述雅特力产品和服务的选择和使用，雅特力概不承担与选择或使用本文所述雅特力产品和服务相关的任何责任。

无论之前是否有过任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为雅特力授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在雅特力的销售条款中另有说明，否则，雅特力对雅特力产品的使用和/或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途(及其依据任何司法管辖区的法律的对应情况)，或侵犯任何专利、版权或其他知识产权的默示保证。

雅特力产品并非设计或专门用于下列用途的产品：(A) 对安全性有特别要求的应用，如：生命支持、主动植入设备或对产品功能安全有要求的系统；(B) 航空应用；(C) 汽车应用或汽车环境；(D) 航天应用或航天环境，且/或(E) 武器。因雅特力产品不是为前述应用设计的，而采购商擅自将其用于前述应用，即使采购商向雅特力发出了书面通知，风险由购买者单独承担，并且独力负责在此类相关使用中满足所有法律和法规要求。

经销的雅特力产品如有不同于本文档中提出的声明和/或技术特点的规定，将立即导致雅特力针对本文所述雅特力产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大雅特力的任何责任。

© 2020 雅特力科技 (重庆) 有限公司 保留所有权利