

32 位微控制器

HC32F120 系列的复位控制器 RMU

适用对象

系列	产品型号
HC32F120	HC32F120H8TA
	HC32F120F8TA
	HC32F120H6TA
	HC32F120F6TA

目 录

1	摘要	3
2	RMU 简介	3
3	HC32F120 系列的 RMU.....	4
3.1	复位方式	4
3.2	复位动作	4
3.3	注意事项	4
3.4	寄存器说明.....	5
4	样例代码	6
4.1	代码介绍	6
4.2	代码运行	8
5	总结	9
6	版本信息 & 联系方式	10

1 摘要

本篇应用笔记主要介绍 HC32F120 系列芯片的复位控制器（RMU）模块，并通过展示 `rmu` 样例代码简要说明如何使用 RMU 模块。

2 RMU 简介

HC32F120 系列的复位控制器（RMU）模块包括上电复位（POR），低电压检测复位（LVDR），端口复位（NRST）在内的 8 种复位源，每个复位源有独立标志位。

3 HC32F120 系列的 RMU

3.1 复位方式

芯片配置了 8 种复位方式。

- 上电复位(POR)
- $\overline{\text{RESET}}$ 引脚复位($\overline{\text{RESET}}$)
- 低电压检测复位 (LVD)
- 专用看门狗复位 (SWDTR)
- 软件复位(SRST)
- M0+ Lockup 复位
- 外部高速振荡器异常停振复位(XTALER)
- RAM 奇偶校验复位

各种复位方式的详细说明请查看本系列芯片用户手册相关章节。

3.2 复位动作

本系列芯片产生复位动作时，会复位芯片内部除 RTC 外设之外的所有资源。RTC 外设资源仅在上电复位被复位。

3.3 注意事项

使用 RMU 模块时，需要注意以下几点：

- 对复位标志寄存器进行写操作前需要通过 PWR_FPRC.PWRWE 标志位写 1 打开写保护。
- 寄存器 PWR_FPRC 的读写方式请查看本系列芯片手册对 PWR_FPRC 寄存器的详细说明。

3.4 寄存器说明

以下为 RMU 模块的寄存器列表，详细寄存器说明请查看本系列芯片用户手册相关章节。

BASE ADDR: 0x40014100

寄存器名	符号	偏移地址	位宽	复位值
复位状态寄存器	RMU_RSTF0	0x00	16	根据不同的复位方式复位值不同

4 样例代码

4.1 代码介绍

用户可写自己的代码来学习验证该模块，也可以直接通过华大半导体的网站下载到设备驱动库（Device Driver Library, DDL）的样例代码并使用其中的 rmu 的样例进行验证。

以下部分简要介绍本 AN 基于 DDL 的 RMU 模块样例 rmu 代码所涉及的各项配置。

1) 复位模式配置

根据需要对 TEST_RESET_MODE 的宏定义进行修改，可以为 RESET_WDT、RESET_XTALERR、RESET_SOFTWARERST、RESET_LVD 之一。

```
/* Test reset mode enum*/
#define RESET_WDT          0u
#define RESET_XTALERR      1u
#define RESET_SOFTWARERST  2u
#define RESET_LVD          3u
/* Modify for test reset mode, can be RESET_WDT,RESET_XTALERR or RESET_SOFTWARERST
or RESET_LVD*/
#define TEST_RESET_MODE    RESET_WDT
```

2) 系统时钟、测试 LED 端口初始化

```
/* Configure system clock. */
SystemClockConfig();
/* configuration uart for debug information */
DDL_UartInit();
DDL_Delay1ms(10ul);
```

3) 读取当前复位源并且通过串口打印，清除复位源寄存器

```
RMU_GetResetCause(&stcResetFlag);
RMU_ClrResetFlag();

PrintResetMode(stcResetFlag);
```

4) 等待 SW2 按键，配置复位功能

```
printf("\nPress SW2 to config reset condition .\n");
/* Wait short press key SW2 */
WaitSw2_ShortPress();

MakeReset();
/* Reset condition configuration finished */
while(1u)
{
    ;
}
```

5) 复位功能配置子函数

```

/**
 * @brief Make reset condition for test
 * @param [in] None
 * @retval None
 */
static void MakeReset(void)
{
    #if (TEST_RESET_MODE == RESET_XTALERR)
        /* Config XTAL status detect error reset on*/
        stc_clk_xtalstd_init_t stcXtalStdCfg;
        stcXtalStdCfg.u8XtalStdState = CLK_XTALSTD_ON;
        stcXtalStdCfg.u8XtalStdMode = CLK_XTALSTD_MODE_RST;
        stcXtalStdCfg.u8XtalStdInt = CLK_XTALSTD_INT_OFF;
        stcXtalStdCfg.u8XtalStdRst = CLK_XTALSTD_RST_ON;
        CLK_XTALStdInit(&stcXtalStdCfg);

        printf("Xtal status detected reset function on.\r\n");
    #elif (TEST_RESET_MODE == RESET_SOFTWARERST)
        /* Software reset MCU */
        printf("Software reset soon.\r\n");
        printf(0);
        NVIC_SystemReset();

    #elif (TEST_RESET_MODE == RESET_WDT)
        /* Configuration Watchdog function */
        stc_swdt_init_t stcSwdtCfg;
        stcSwdtCfg.u32CountCycle = SWDT_COUNTER_CYCLE_256;
        stcSwdtCfg.u32ClockDivision = SWDT_CLOCK_DIV1;
        stcSwdtCfg.u32RefreshRange = SWDT_RANGE_100PCT;
        stcSwdtCfg.u32LPModeCountEn = Disable;
        stcSwdtCfg.u32RequestType = SWDT_TRIG_EVENT_RESET;

        SWDT_Init(&stcSwdtCfg);
        printf("Watchdog reset function on.\r\n");
        SWDT_ReloadCounter();
    #elif (TEST_RESET_MODE == RESET_LVD)
        stc_pwc_lvd_cfg_t stcPwcLvdCfg;

        /* Config LVD */
        stcPwcLvdCfg.u16IRDIS = PWC_LVD_IR_ON;
        stcPwcLvdCfg.u16IRSel = PWC_LVD_RST;
        stcPwcLvdCfg.u16NMISel = PWC_LVD_INT_MASK;
        stcPwcLvdCfg.u16Level = PWC_LVD_Level2;
        stcPwcLvdCfg.u16DFDIS = PWC_LVD_DF_OFF;
        stcPwcLvdCfg.u16DFSel = PWC_LVD_DFS_2;

        PWC_LvdLevelConfig(&stcPwcLvdCfg);

        /* Enable LVD */
        PWC_LvdCmd(Enable);
        /* Enable cmp result output */
        PWC_LvdCmpOutputCmd(Enable);

        printf("LVD reset on, please make the power supply voltage lower than 3.0V\r\n");
    #endif
}

```

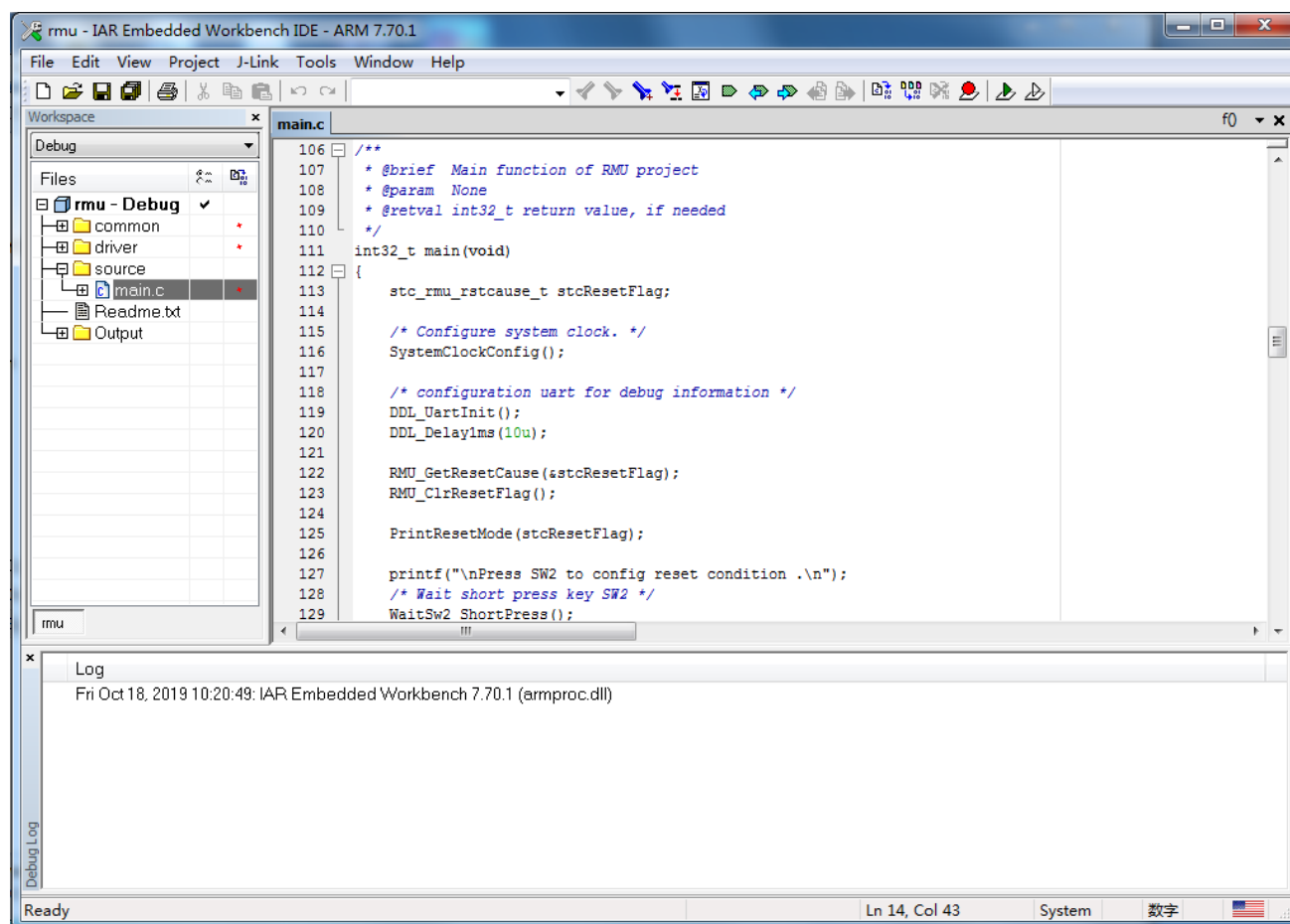
4.2 代码运行

用户可以通过华大半导体的网站下载到 HC32F120 的 DDL 的样例代码（rmu），并配合评估用板（STK-HC32F120-LQFP44-080-V11）运行相关代码学习使用 RMU 模块。

以下部分主要介绍如何在评估板上运行 rmu 样例代码并观察结果：

- 确认安装正确的 IAR EWARM v7.7 工具（请从 IAR 官方网站下载相应的安装包，并参考用户手册进行安装）。
- 从华大半导体网站下载 HC32F120 DDL 代码。
- 下载并运行 rmu 样例的工程文件：

1) 打开 rmu 工程，并打开 ‘main.c’ 如下视图：




2) 选择复位模式，如选择 SWDT 复位则定义 TEST_RESET_MODE 为 RESET_WDT。点



击重新编译整个项目。

3) 电脑通过 USB micro 线连接开发板的 USB 口 J1。电脑打开串口调试软件打开相应端口，用于查看代码运行的串口输出信息。

- 4) 点击  将代码下载到评估板上，全速运行。
- 5) 根据串口信息提示 “Press SW2 to config reset condition” 时，短按按键 SW2。
- 6) 自动或手动产生复位条件，根据串口提示查看复位情况。

5 总结

以上章节简要介绍了 HC32F120 系列的 RMU，说明了 RMU 模块的寄存器及注意事项，并且演示了如何使用 `rmu` 样例代码，在实际开发中用户可以根据自己的需要配置和使用 RMU 模块。

6 版本信息 & 联系方式

日期	版本	修改记录
2019/10/31	Rev1.0	初版发布



如果您在购买与使用过程中有任何意见或建议，请随时与我们联系。

Email: mcu@hdsc.com.cn

网址: www.hdsc.com.cn

通信地址: 上海市浦东新区中科路 1867 号 A 座 10 层

邮编: 201203

