



GMF03x 勘误手册

版本：1.0

发布日期：2020-04-28

版权所有 © 2020 深圳市汇顶科技股份有限公司。保留一切权利。

非经本公司书面许可，任何单位和个人不得对本手册内的任何部分擅自摘抄、复制、修改、翻译、传播，或将其全部或部分用于商业用途。

商标声明

GOODIX 和其他汇顶商标均为深圳市汇顶科技股份有限公司的商标。本文档提及的其他所有商标或注册商标，由各自的所有人持有。

免责声明

本文档中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。

深圳市汇顶科技股份有限公司（以下简称“GOODIX”）对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。GOODIX 对因这些信息及使用这些信息而引起的后果不承担任何责任。

未经 GOODIX 书面批准，不得将 GOODIX 的产品用作生命维持系统中的关键组件。在 GOODIX 知识产权保护下，不得暗或以其他方式转让任何许可证。

深圳市汇顶科技股份有限公司

总部地址：深圳市福田区腾飞工业大厦 B 座 2 层、13 层

电话：+86-755-33338828 传真：+86-755-33338830

网址：<http://www.goodix.com>

前言

编写目的

本文档描述 GMF03x 系列芯片的使用限制场景。

阅读本文档时，可参考《GMF03x 用户手册》与《GMF03x 数据手册》。

读者对象

本文适用于以下读者：

- 嵌入式开发工程师
- 芯片选型人员
- 芯片测试人员
- 产品经理

版本说明

本文档为第 1 次发布，对应的产品系列为 GMF03x。

修订记录

版本	日期	修订内容
1.0	2020-04-28	首次发布。

目录

前言	1
1 芯片使用限制描述	1
1.1 RTC	1
1.1.1 非 Bypass 方式下 (RTC_CTL 寄存器的 BYP=0) 两断点时间间隔小于 1s 时, RTC 的影子寄存器更新异常	1
1.2 DEBUG	1
1.2.1 Debug 模式下, 芯片退出 Standby 模式后, 断点配置信息丢失	1
1.3 SPI	1
1.3.1 MCU 作为 SPI 主设备, 在 TI 协议单接收模式下 (SPI_CTL0 寄存器的 SDUPDIR=1), 当前数据帧接收完成后, 进行禁能 SPI 操作 (SPI_CTL0 寄存器的 SPIEN=0), 主设备可能会多发送一个异常时钟, 导致后续帧传输错位	1
1.4 EXTI	2
1.4.1 在 NVIC 中使能 I2C 中断, EXTI 中禁能 I2C 中断请求 (EXTI_INTMSK 寄存器的 INTMSK23=0) 的情况下, 若 MCU 通过 WFI 方式进入 Stop 模式, MCU 将被异常唤醒	2

1 芯片使用限制描述

使用 GMF03x 系列芯片的 RTC、DEBUG、SPI 及 EXTI 外设模块时存在一定的限制条件。

下面章节将详细描述这些限制条件及其规避方案。

1.1 RTC

1.1.1 非 Bypass 方式下（RTC_CTL 寄存器的 BYP=0）两断点时间间隔小于 1s 时，RTC 的影子寄存器更新异常

- 描述

在 Debug 模式下，设置 RTC 计时暂停（DBG_FRZO 寄存器的 RTCSTP=1），然后使用非 Bypass 方式（RTC_CTL 寄存器的 BYP=0）读取日历，且两次断点时间间隔小于 1s 时，RTC 影子寄存器（RTC_SS 寄存器、RTC_TIME 寄存器和 RTC_DATE 寄存器）的值可能存在不更新的情况。

- 规避方案

使用 Bypass 方式（RTC_CTL 寄存器的 BYP=1）读取 RTC 的 RTC_SS 寄存器、RTC_TIME 寄存器和 RTC_DATE 寄存器。

1.2 DEBUG

1.2.1 Debug 模式下，芯片退出 Standby 模式后，断点配置信息丢失

- 描述

在 Debug 模式下，当芯片退出 Standby 模式后，断点单元会自动复位。所以即使在使能 Standby 调试功能（DBG_CTL 寄存器的 STB=1）的情况下，断点配置信息也会丢失。

- 规避方案

芯片退出 Standby 模式后手动添加断点。

1.3 SPI

1.3.1 MCU 作为 SPI 主设备，在 TI 协议单接收模式下（SPI_CTL0 寄存器的 SDUPDIR=1），当前数据帧接收完成后，进行禁能 SPI 操作（SPI_CTL0 寄存器的 SPIEN = 0），主设备可能会多发送一个异常时钟，导致后续帧传输错位

- 描述

MCU 作为 SPI 主设备，在 TI 协议单接收模式下（SPI_CTL0 寄存器的 SDUPDIR=1），当前数据帧接收完成后，立即执行禁能 SPI 操作（SPI_CTL0 寄存器的 SPIEN = 0），主设备可能会多发出一个异常时钟。该过程中主设备会接收一个 bit 数据，从而使从设备进入下一帧数据的发送状态。这将导致主设备在接收下一帧数据时，从设备数据传输出现异常。

- 规避方案

方案 1: MCU 作为 SPI 主设备, 在 TI 协议下, 配置为全双工模式 (SPI_CTL0 寄存器的 HFDUPEN=0)。

方案 2: 完成一次数据传输后, 启动下一次数据传输前, 复位从设备 SPI 模块。

1.4 EXTI

1.4.1 在 NVIC 中使能 I2C 中断, EXTI 中禁能 I2C 中断请求 (EXTI_INTMSK 寄存器的 INTMSK23=0) 的情况下, 若 MCU 通过 WFI 方式进入 Stop 模式, MCU 将被异常唤醒

- 描述

在 NVIC 中使能 I2C 中断, EXTI 中禁能 I2C 中断请求 (EXTI_INTMSK 寄存器的 INTMSK23=0) 的情况下, MCU 通过 WFI 方式进入 Stop 模式, 当 I2C 模块接收到匹配地址时, I2C 将产生一个地址匹配中断, 从而使 MCU 被唤醒。

- 规避方案

在 MCU 通过 WFI 方式进入 Stop 模式前, 禁能 MCU 从 Stop 模式唤醒 (I2C_CTL0 寄存器的 WAKEUPEN=0), 从而关闭 I2C 地址匹配唤醒功能。