

文档编号: AN2002

上海东软载波微电子有限公司

应用笔记

ES32F065x

修订历史

版本	修订日期	修改概要
V1.0	2018-12-14	初版
V1.1	2019-03-25	1. 温度传感器简写 TEMP 改写为 TSENSE; 2. 变更 Logo。
V1.2	2019-09-10	1. 增加 FLASH 读保护部分; 2. 增加配置字部分注意事项; 3. 增加 CAN 模块应用注意事项。
V1.3	2019-12-23	1. 增加 TSENSE 应用注意; 2. 增加系统时钟选择章节相关注意事项。
V1.4	2020-04-27	1. 增加电源模块应用注意; 2. 增加 CAN 模块应用注意。
V1.5	2020-09-29	1. 增加 ADC 模块应用注意; 2. 增加低功耗模式应用注意。 3. 细化电源模块应用注意。
V1.6	2020-11-20	1. 增加外部晶振应用注意; 2. 增加 USART 模块应用注意; 3. 增加 UART 模块应用注意; 4. 增加 GPIO 模块应用注意; 5. 增加 POR 模块应用注意。
V1.7	2021-05-18	1. 调整文档目录结构; 2. 增加 GPIO 模块应用注意; 3. 增加 UART 模块应用注意; 4. 增加 ADC 模块应用注意; 5. 增加 CAN 模块应用注意; 6. 增加低功耗模块应用注意; 7. 增加 5V 电路供电应用注意。

地 址：中国上海市龙漕路 299 号天华信息科技园 2A 楼 5 层

邮 编：200235

E-mail: support@essemi.com

电 话：+86-21-60910333

传 真：+86-21-60914991

网 址：http://www.essemi.com/

版权所有©

上海东软载波微电子有限公司

本资料内容为上海东软载波微电子有限公司在现有数据资料基础上慎重且力求准确无误编制而成，本资料中所记载的实例以正确的使用方法和标准操作为前提，使用方在应用该等实例时请充分考虑外部诸条件，上海东软载波微电子有限公司不承担或确认该等实例在使用方的适用性、适当性或完整性，上海东软载波微电子有限公司亦不对使用方因使用本资料所有内容而可能或已经带来的风险或后果承担任何法律责任。基于使本资料的内容更加完善等原因，上海东软载波微电子有限公司保留未经预告的修改权。使用方如需获得最新的产品信息，请随时用上述联系方式与上海东软载波微电子有限公司联系

目 录

内容目录

第 1 章	概述	5
1.1	开发环境	5
1.2	库函数选择	5
1.3	寄存器写保护	5
1.3.1	系统写保护	5
1.3.2	RTC 写保护	5
1.3.3	TSENSE 写保护	5
1.3.4	IWDT 写保护	5
1.3.5	WWDT 写保护	6
1.4	写 1 清零寄存器	6
1.5	位带操作	6
第 2 章	系统控制	7
2.1	系统时钟选择	7
2.1.1	内部高速 24MHz(默认时钟)	7
2.1.2	外部时钟 HOSC(4~24MHz)	7
2.1.3	48MHz(使用 HRC 倍频)	7
2.1.4	48MHz(使用 HOSC 倍频)	7
2.1.5	外部低速时钟(LOSC)	7
2.2	外部晶振	8
2.3	IAP 操作程序	8
2.4	FLASH 读保护	8
2.5	低功耗模式	8
2.6	电源管理模块	8
2.7	配置字	9
第 3 章	外设	10
3.1	GPIO 模块	10
3.2	CAN 模块	10
3.3	I2C 模块	10
3.4	USART 模块	10
3.5	UART 模块	10
3.6	PIS 模块	11
3.7	SPI 模块	11
3.8	TSENSE 模块	11
3.9	ADC 模块	11
第 4 章	最小系统电路	12
4.1	3.3V 供电系统	12
4.1.1	LQFP100 封装芯片最小系统电路	12
4.1.2	LQFP64 封装芯片最小系统电路	13
4.1.3	LQFP48/QFN48 封装芯片最小系统电路	14
4.2	5V 供电系统	15
4.2.1	LQFP100 封装芯片最小系统电路	15

4.2.2	LQFP64 封装芯片最小系统电路	16
4.2.3	LQFP48/QFN48 封装芯片最小系统电路.....	17

第1章 概述

1.1 开发环境

推荐用户使用 Keil5 、IAR8.11 或者 iDesigner 进行固件开发。由于 Keil4 不支持 PACK 机制，故不推荐用户使用 Keil4。

1.2 库函数选择

ES32 系列芯片提供 2 种类型库函数 ALD 和 MD：

ALD：提供较为完善的封装，提供更为人性化的 API，适合大部分用户；

MD：基本上只提供寄存器位域级别的“读”、“写”接口，适合对芯片底层较为熟悉的用户。

如果用户对速度不是要求非常严格，一般情况下推荐用户使用 ALD 库。可以减少用户学习时间，增加代码可移植性，最终缩短用户产品的开发周期。

1.3 寄存器写保护

为避免程序的异常导致运行错误，芯片写保护寄存器用于阻止对被保护的寄存器误操作。

系统控制单元，GPIO，RTC，WDT 等模块支持寄存器写保护，对被保护的寄存器进行写之前需要解除写保护状态（允许写），否则无法对写保护寄存器写入。操作完成后，再使能写保护（禁止写）。库函数中均提供相应宏定义进行解除保护和使能保护。

1.3.1 系统写保护

系统控制寄存器的访问操作会影响整个芯片的运行状态，芯片提供系统设置保护寄存器 SYSCFG_PROT。对 SYSCFG_PROT 寄存器以字方式写入 0x55AA6996 会解除写保护，对该寄存器写入其他任何值都会使能写保护。

可通过读 SYSCFG_PROT 寄存器确认写保护状态，读出值为 0x1，表示当前处于写保护状态；读出值为 0x0 表示当前处于写保护解除状态。

SYSCFG_PROT 保护的寄存器为除 SYSCFG_PROT 寄存器外的 SYSCFG、PMU、CMU、RMU 模块所有寄存器。

1.3.2 RTC写保护

对 RTC_WPR 寄存器以字方式写入 0x55AAAA55 会解除写保护，写入其他值使能写保护。

可通过读 RTC_WPR 寄存器确认 RTC 模块是否处于写保护状态，读出值为 0x1，表示当前处于写保护状态；读出值为 0x0 表示 RTC 模块处于写保护解除状态。

该寄存器保护除自身外的 RTC 所有寄存器。

1.3.3 TSENSE写保护

对 TSENSE_WPR 寄存器以字方式写入 0xA55A9669 会解除写保护，写入其他值使能写保护。

可通过读 TSENSE_WPR 寄存器确认 TSENSE 模块是否处于写保护状态，读出值为 0x1，表示当前处于写保护状态；读出值为 0x0 表示 TSENSE 模块处于写保护解除状态。

该寄存器保护除自身外的 TSENSE 所有寄存器。

1.3.4 IWDWT写保护

对 IWDWT_LOCK 寄存器以字方式写入 0x1ACCE551 会解除写保护，写入其他值使能写保护。

可通过读 IWDWT_LOCK 寄存器确认 IWDWT 模块是否处于写保护状态，读出值为 0x1，表示当前处于写保护状态；读出值为 0x0 表示 IWDWT 模块处于写保护解除状态。

该寄存器保护除自身外的 IWDT 所有寄存器。

1.3.5 WWDT写保护

对 WWDT_LOCK 寄存器以字方式写入 0x1ACCE551 会解除写保护，写入其他值使能写保护。

可通过读 WWDT_LOCK 寄存器确认 WWDT 模块是否处于写保护状态，读出值为 0x1，表示当前处于写保护状态；读出值为 0x0 表示 WWDT 模块处于写保护解除状态。

该寄存器保护除自身外的 WWDT 所有寄存器。

1.4 写 1 清零寄存器

中断标志寄存器都是用“写 1 清零”的方式来操作。对于“写 1 清零”的寄存器，不可使用“读-修改-写”的方式来进行“写 1 清零”，否则会引起标志误清，进而产生漏中断的后果。对该类寄存器操作需要以字方式进行写。

例：清除 DMA 模块通道 0 的中断标志：

正确写法：DMAx->ICFR = 1;

错误写法：DMAx->ICFR |= 1;

1.5 位带操作

位带扩展区把每个 bit 扩展为一个 32-bits 的字，通过访问这些字可达到访问原始 bit 的目的；某个 bit 所在字的地址为 A，位序号为 N($0 \leq N \leq 31$)，则该 bit 位带扩展后的地址为：

SRAM: AliasAddr=0x22000000+(A-0x20000000)x32+Nx4

外设: AliasAddr=0x42000000+(A-0x40000000)x32+Nx4

库函数中提供位带操作 API：

RAM 位带: void BITBAND_SRAM(uint32_t *addr, uint32_t bit, uint32_t val);

外设位带: void BITBAND_PER(volatile uint32_t *addr, uint32_t bit, uint32_t val);

第2章 系统控制

2.1 系统时钟选择

系统上电默认使用内部 24MHz 高速时钟(HRC)作为系统时钟。

若系统运行 48MHz/32MHz 主频，建议将 APB2 总线进行 2 分频，否则可能会导致低速外设运行不正常，分频操作建议使用 ALD 库中 API: `ald_cmu_div_config(CMU_PCLK_2, CMU_DIV_2)`;
几种常用系统时钟配置：

2.1.1 内部高速 24MHz(默认时钟)

此种系统时钟不需要用户做任何配置。

2.1.2 外部时钟HOSC(4~24MHz)

外部高速时钟要求为 4MHz 的倍数，如：4MHz、8MHz、12MHz、16MHz。

首先要确认焊接了外部高速时钟，并已知外部高速时钟的频率，假如外部高速时钟为 12MHz，则配置方式如下：

```
ald_cmu_clock_config(CMU_CLOCK_HOSC, 12000000);
```

2.1.3 48MHz(使用HRC倍频)

配置方式如下：

```
ald_cmu_pll1_config(CMU_PLL1_INPUT_HRC_6, CMU_PLL1_OUTPUT_48M);
```

```
ald_cmu_clock_config(CMU_CLOCK_PLL1, 48000000);
```

2.1.4 48MHz(使用HOSC倍频)

外部高速时钟要求为 4MHz 的倍数，如：4MHz、8MHz、12MHz、16MHz。

首先要确认焊接了外部高速时钟，并已知外部高速时钟的频率，假如外部高速时钟为 12MHz，则配置方式如下：

```
ald_cmu_pll1_config(CMU_PLL1_INPUT_HOSC_3, CMU_PLL1_OUTPUT_48M);
```

```
ald_cmu_clock_config(CMU_CLOCK_PLL1, 48000000);
```

2.1.5 外部低速时钟(LOSC)

首先要确认焊接了外部低速时钟，配置方式如下：

```
ald_cmu_clock_config(CMU_CLOCK_LOSC, 32768);
```

需要注意的是，当系统时钟配置为低速时钟时(低于 1MHz)，SysTick 中断将会被迫关闭。ALD 提供的延迟类函数禁止使用。

2.2 外部晶振

外部高速振荡器的典型应用连接：

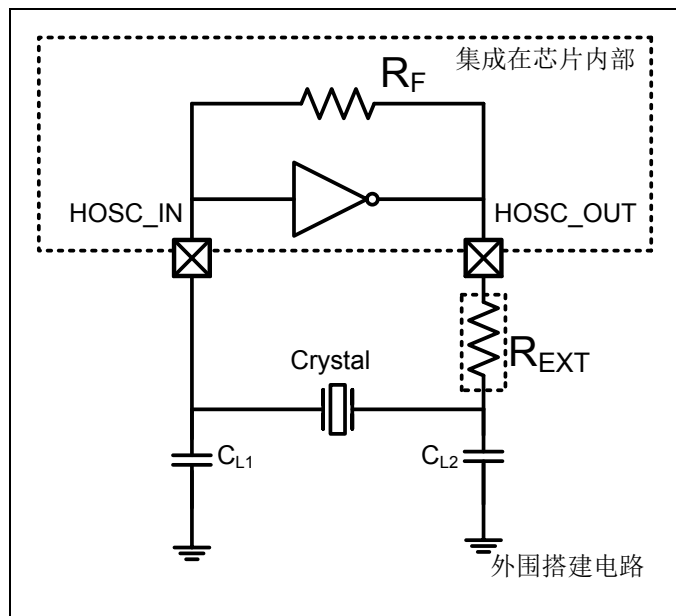


图 1-1 外部高速振荡器连接图

注 1: R_{EXT} 为可选电阻，其阻值取决于具体晶振规格特性；

注 2: 当晶振频率不低于 16MHz 时，需选用 ESR 不高于 30Ω 的晶振，且 C_{L1}, C_{L2} 容值需小于 5pF。

2.3 IAP操作程序

芯片内置 IAP 自编程固化模块，由硬件电路实现。推荐使用 IAP 方式对 FLASH 进行擦、写操作，可以减少用户代码量。

如果不使用 IAP 操作 FLASH，则需要将操作 FLASH 部分的代码放到 SRAM 上执行，具体方法可参考对应例程。

2.4 FLASH读保护

规则描述：当 FLASH 的读保护级别设置为 level1 或 level2 时，运行在 SRAM 上的程序不能有读 FLASH 操作。

典型应用 1：运行在 SRAM 上的程序想读取 FLASH。将读 FLASH 操作放在 FLASH 上执行；

典型应用 2：程序运行在 SRAM 上，响应中断请求。将中断向量表拷贝至 SRAM 中，并设置中断向量偏移地址（SYSCFG_VTOR 和 SYSCFG_MRMP.VTOEN）

2.5 低功耗模式

注意事项 1：从 Standby 模式下唤醒时，会清除 RTC 模块寄存器。即从 Standby 唤醒后，RTC 时间需要重新初始化。其中备份域 128Bytes 的 SRAM 不受影响，仍保持进入 Standby 之前的值。

注意事项 2：进入 STOP1/STOP2 模式之前，置位 CR.LPSTOP 位可降低 STOP 功耗。

2.6 电源管理模块

注意事项 1：芯片电源推荐+3.3V 供电系统。若采用 5V 供电系统，且电源纹波较大，则必须进行特殊防护，防护方法参考 4.2 章节。

注意事项 2: 由于部分芯片 POR 的默认值达到 2.8V，故应用系统中应确保在芯片上电期间，VDD 大于 2.8V。

2.7 配置字

注意事项 1: 使用 ESLink II/ESLink II mini 修改芯片配置字之后，芯片需要断电后重新上电，才能正常调试。

注意事项 2: 使用 ESBurner 擦除芯片后，需要执行配置字编程，否则将导致芯片运行不正常。即点击“擦除”按钮后，需要点击“配编”按钮。

第3章 外设

3.1 GPIO模块

注意事项 1: 未使用的 GPIO 管脚建议设置为输出固定电平并悬空，若设置为输入，须加上拉或下拉电阻接到电源或地；

注意事项 2: 备份域 IO(PC13/PC14/PC15)禁止高速翻转，翻转频率不能超过 1Hz。

注意事项 3: 外部中断的有效电平宽度要大于 100uS；

注意事项 4: IO 的中断延时约 200uS，使用 IO 中断唤醒低功耗模式(STOP1/STOP2/STANDBY)时，其唤醒时间也需要增加 200uS 左右；

注意事项 5: 使用 IO 触发外部中断时，触发信号频率不超过 2.5KHz；

注意事项 6: PA9/PA10/PB3 在系统启动期间有一个高电平的脉冲，宽度约为 20mS。

3.2 CAN模块

注意事项 1: 当 MCU 使用中断方式接收，通信速率是 1Mbps/s，并且帧间隔小于 30uS 时，则 MCU 的主频应运行在 48MHz，否则会有丢帧风险。

注意事项 2: 通信速率范围 10Kbits/s--1Mbps/s。当速率低于 20Kbits/s 时，要求 PCLK1 频率不超过 24MHz；

注意事项 3: 当接收到一帧数据时，需要检查 0x4000B030 地址 bit21，若该位被置位，则该帧可能为错误帧，应将对应 RXFIFO 中的数据全部丢弃。具体实现可参考 ALD 库函数中 __can_rx_check() 函数。

注意事项 4: 由于 CAN 标准中要求时钟精度在 1.58%以内，故当产品的使用环境比较恶劣时，建议使用外部高精度晶振。外部晶振的频率可以使用 4MHz、8MHz、12MHz 或 16MHz，推荐使用 12MHz(和 ES-PDS 板保持一致)。

3.3 I2C模块

注意事项 1: I2C 工作在主机模式时，若从机一直拉低时钟线，主机可以通过关闭 I2C 模块来释放时钟线；

注意事项 2: I2C 作为主机与外部 EEPROM 进行通信，若 SDA 线一直被 EEPROM 器件拉低，可以通过以下 2 种方式解除死锁现象：

1. 通过重新给 EEPROM 上电解除死锁

2. 主机 SCL 线模拟输出 9 个时钟信号后，再模拟发送一个停止信号，EEPROM 会终止此次通信并释放 SDA 线。

3.4 USART模块

注意事项 1: IDLE 中断需要在接收到一个数据之后再打开，否则该中断会立即触发，并且中断标志不能被清除；

注意事项 2: RX 波特率误差容忍度(+0.1%，-6%)；在 USART 初始化时，建议将波特率上浮+1.5%左右。

3.5 UART模块

注意事项 1: 在 LIN 模式下，SR.BF 标志无效，可通过 RIF.LINBKIF 标志判断总线是否出现断开符；

注意事项 2: TC 中断在每发送完一个字节后均被置起，而不是在数据帧发送完成后被置起，所以不

能通过 RIF.TCIF 标志判断一帧是否发生完成。可通过判断 SR.TEM 标志，来确定一帧数据是否发送完成；

注意事项 3：通过 DMA 发送帧数据时，一帧数据的第一个字节需要直接写 TBR 寄存器，后续字节再使用 DMA 方式发送。具体实现可参考 SDK 中对应例程；

注意事项 4：配合 DMA 使用时，DMA 的 burst 必须为 1，R_power 必须为 0；

注意事项 5：在中断模式下，发送时若向 TX_FIFO 中写入多个数据，则每次写数据前需确认 FIFO 是否满，避免 FIFO 上溢；接收时需要在接收中断函数中将 RX_FIFO 读空。该项应用注意在 ALD 库中已经做了处理，使用 MD 库时需自行处理；

注意事项 6：当发生溢出错误时，需要将 RX_FIFO 进行复位。该项应用注意在 ALD 库中已经做了处理，使用 MD 库时需自行处理；

注意事项 7：使用中断方式发送数据时，需要按照如下顺序进行：先清除 TXS 中断，再向 TBR 寄存器写入数据。

3.6 PIS模块

注意事项 1：PIS 模块不支持 TIMER 的触发事件，控制另外一个 TIMER；

注意事项 2：使用 TIMER 的输入捕捉事件触发 ADC 转换时，仅支持 TIMER 的通道 4，其余通道不支持该用法。

3.7 SPI模块

注意事项 1：SPI 在只读模式下 (CON1.RXO 置位)，STAT.BUSY 标志会一直被置起，故不能通过该位判断通信是否完成。若需要清除 STAT.BUSY 标志，需要将 CON1.RXO 清零。

注意事项 2：从机模式下，当片选使用软件控制时 (SPI_CON1.SSEN=1)，无论 SPI_CON1.SSOUT 是 0 还是 1，SPI 都可以正常收发数据。即在这种应用条件下 SPI_CON1.SSOUT 起不到控制作用。

3.8 TSENSE模块

注意事项 1：该模块在使用之前需进行校准，校准方法可参考 ALD 库中 ald_tsense_init() 函数；

3.9 ADC模块

注意事项 1：ADC 不支持连续模式。即转换完一组序列后 (标准序列或插入序列)，需重新触发 (软件触发或 PIS 触发)。库函数为了保持和前期版本的兼容性，ADC 初始化结构体的成员变量 cont 仍然保持，但不起作用。

注意事项 2：根据采样频率 (PCLK2/2^CCR.CKDIV) 设置 ADC 的运行模式，当采样频率小于 100KHz 时，可以使用低速模式 (CCR.PWRMODSEL=0) 减小 ADC 模块运行功耗；当采样频率大于 100KHz 时，必须使用高速模式 (CCR.PWRMODSEL=1)；

注意事项 3：对 ADC 绝对精度有较高要求的应用系统，可通过软件补偿 (Gain/Offset) 的方式提高精度，具体实现可参考 ALD 库中 ald_adc_normal_get_value_calib() 函数。同时可以使用内部 2.0V 参考源去标定其余采样通道的电压值：设待测通道的 ADC 值为 value0，通道 18 (内部 2.0V) 的 ADC 值为 value1，则待测通道的电压值为： $V(mV) = ((value0 - offset) \times 2000) / (value1 - offset)$

第4章 最小系统电路

4.1 3.3V供电系统

4.1.1 LQFP100 封装芯片最小系统电路

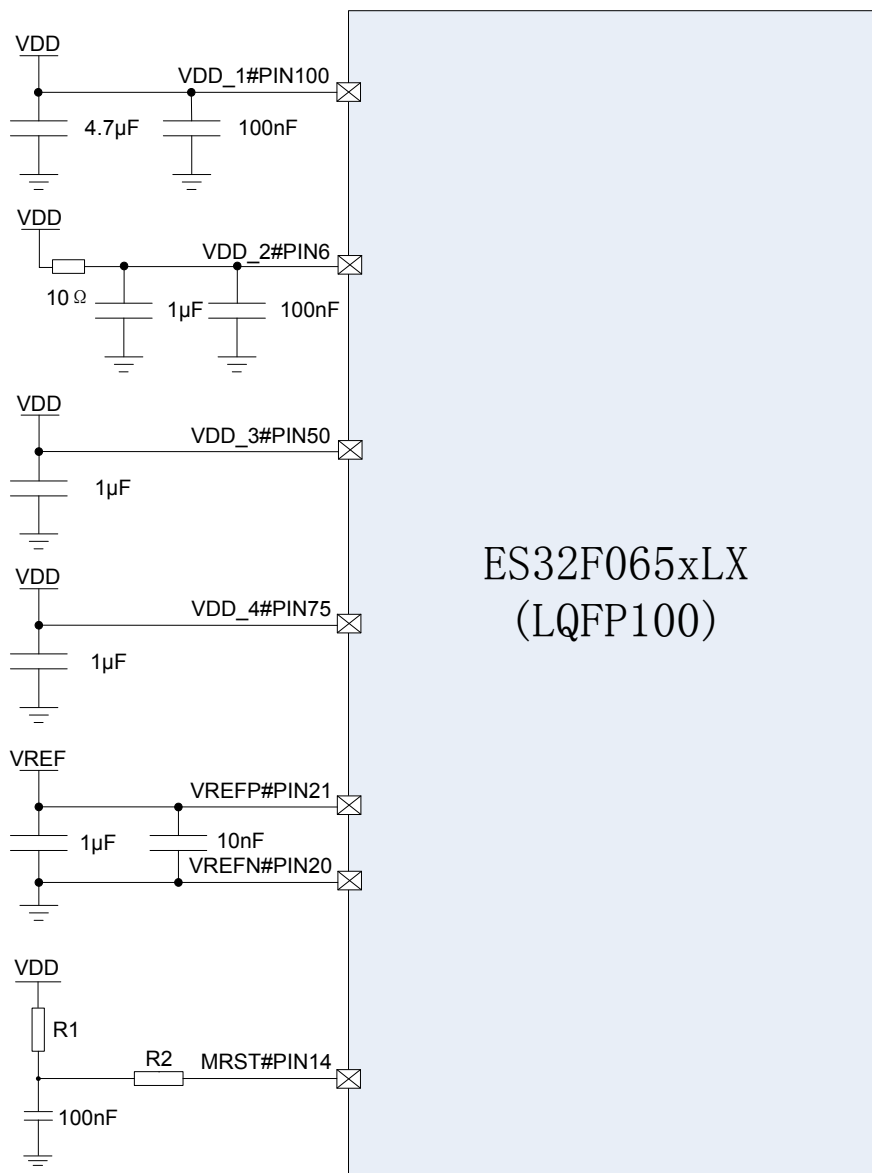


图 4-1 LQFP100 封装芯片最小系统电路

注 1: 每一组电源必须连接如图所示的陶瓷耦合电容。这些电容必须尽可能地靠近芯片的相应管脚, 才能保证芯片的运行性能。

注 2: PIN100 是芯片主电源, PIN6 是备份域电源, 需特殊防护。

注 3: VREFP 和 VREFN 管脚在复用为 ADC 外部参考时才需要外接电容。

注 4: MRST 引脚采用 RC 复位, 其中 $47\text{K}\Omega \leq R1 \leq 100\text{K}\Omega$, 电容 $C1=100\text{nF}$, $R2$ 为限流电阻, $0.1\text{K}\Omega \leq R2 \leq 1\text{K}\Omega$ 。

4.1.2 LQFP64 封装芯片最小系统电路

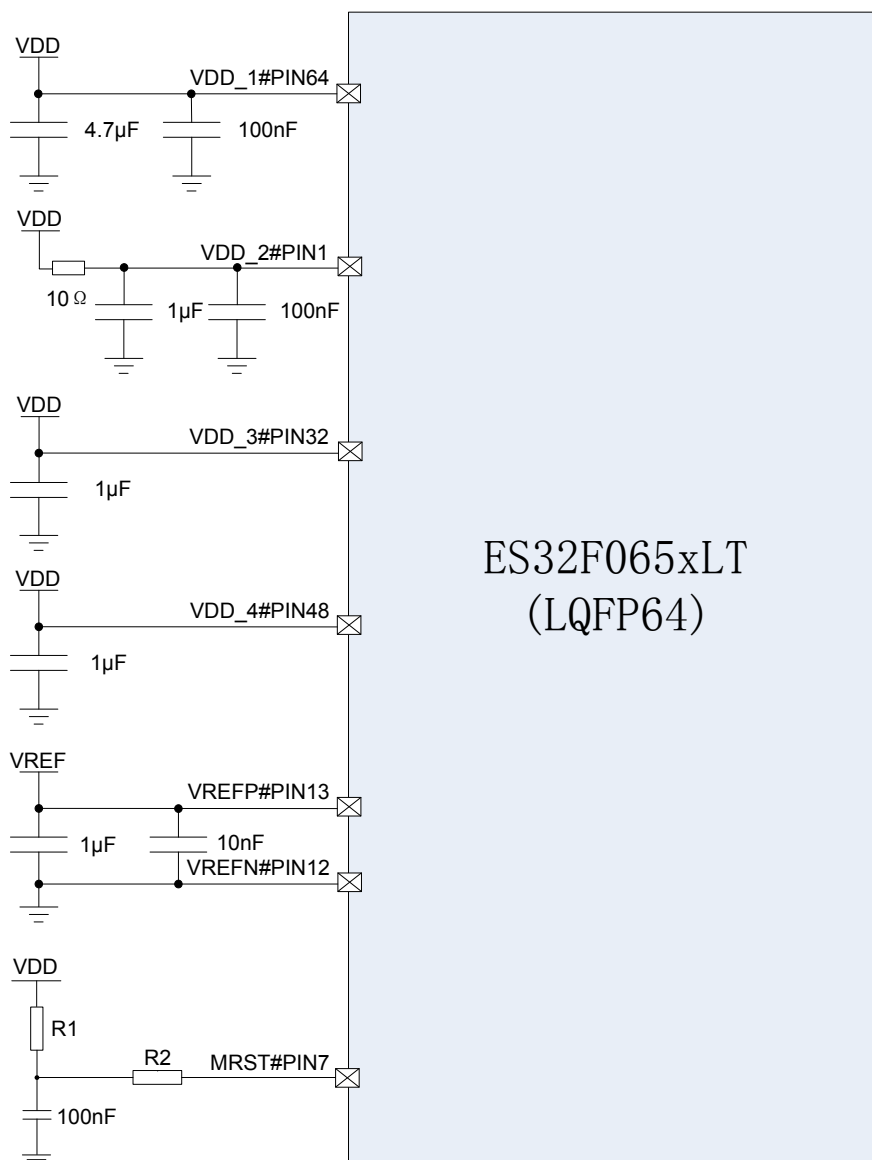


图 4-2 LQFP64 封装芯片最小系统电路

注 1: 每一组电源必须连接如图所示的陶瓷耦合电容。这些电容必须尽可能地靠近芯片的相应管脚, 才能保证芯片的运行性能。

注 2: PIN64 是芯片主电源, PIN1 是备份域电源, 需特殊防护。

注 3: VREFP 和 VREFN 管脚在复用为 ADC 外部参考时才需要外接电容。

注 4: MRST 引脚采用 RC 复位, 其中 $47\text{K}\Omega \leq R1 \leq 100\text{K}\Omega$, 电容 $C1=100\text{nF}$, $R2$ 为限流电阻, $0.1\text{K}\Omega \leq R2 \leq 1\text{K}\Omega$ 。

4.1.3 LQFP48/QFN48 封装芯片最小系统电路

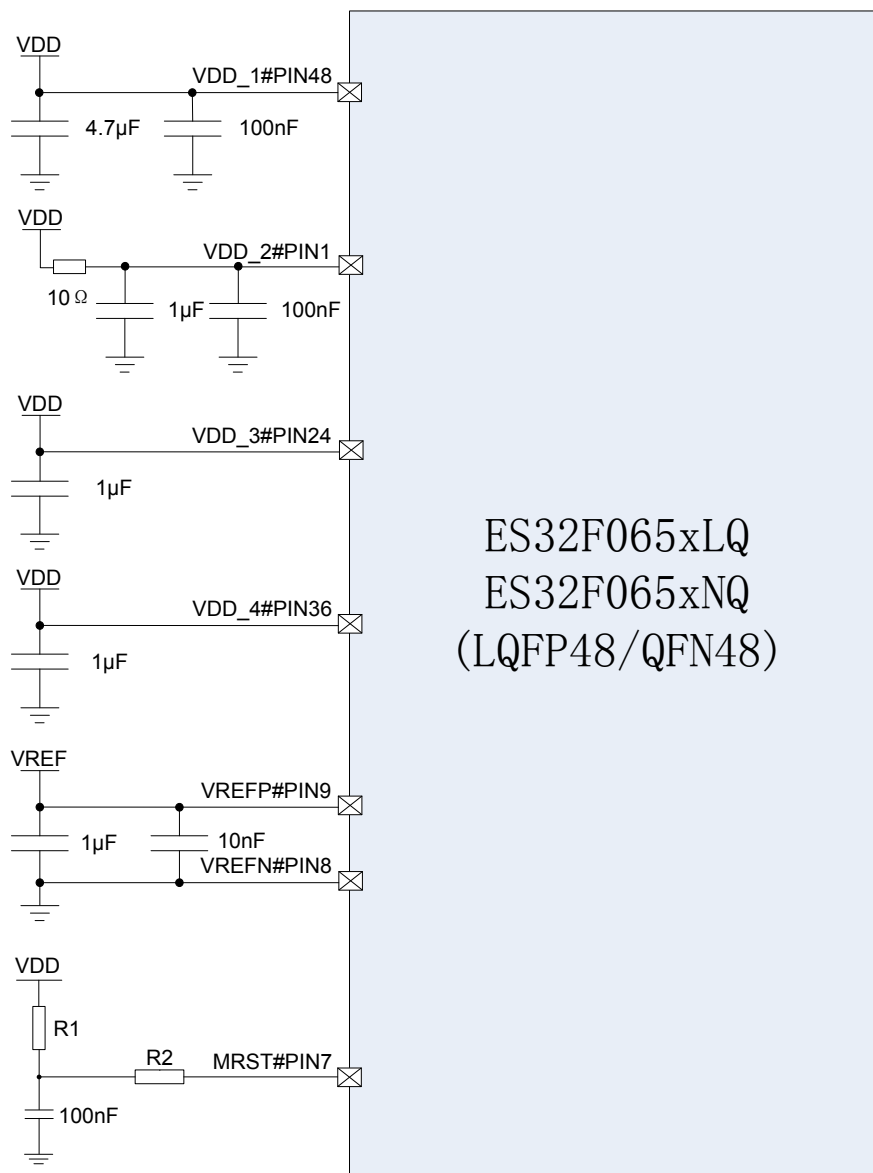


图 4-3 LQFP48/QFN48 封装芯片最小系统电路

- 注 1: 每一组电源必须连接如图所示的陶瓷耦合电容。这些电容必须尽可能地靠近芯片的相应管脚, 才能保证芯片的运行性能。
- 注 2: PIN48 是芯片主电源, PIN1 是备份域电源, 需特殊防护。
- 注 3: VREFP 和 VREFN 管脚在复用为 ADC 外部参考时才需要外接电容。
- 注 4: MRST 引脚采用 RC 复位, 其中 $47\text{K}\Omega \leq R1 \leq 100\text{K}\Omega$, 电容 $C1=100\text{nF}$, R2 为限流电阻, $0.1\text{K}\Omega \leq R2 \leq 1\text{K}\Omega$ 。

4.2 5V供电系统

当芯片电源电压 VDD 大于 3.3V 时，建议在芯片与系统供电电源之间串接一颗 6.8Ω 电阻，示意图如下：

4.2.1 LQFP100 封装芯片最小系统电路

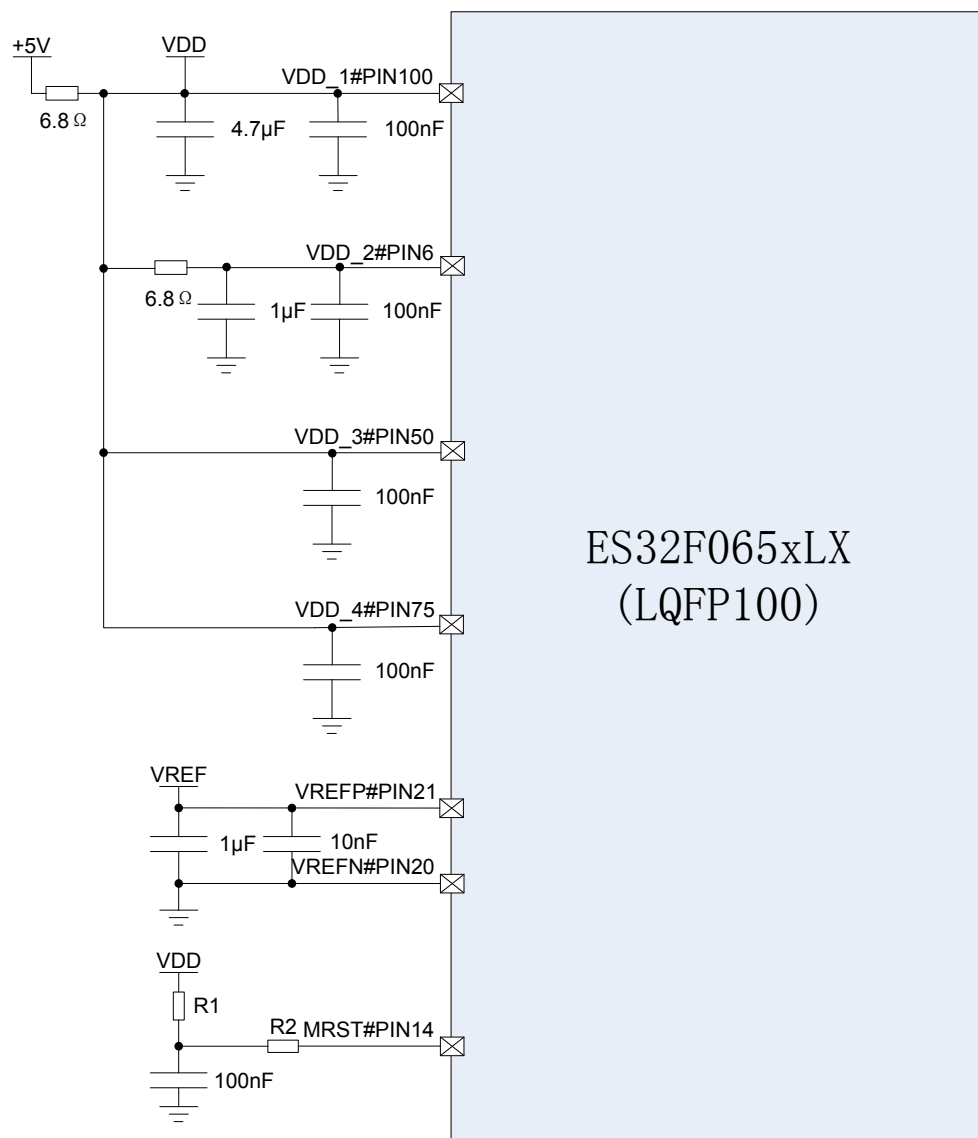


图 4-4 LQFP100 封装芯片电源串接电阻示意图

注 1：每一组电源必须连接如图所示的陶瓷耦合电容。这些电容必须尽可能地靠近芯片的相应管脚，才能保证芯片的运行性能。

注 2：PIN100 是芯片主电源，PIN6 是备份域电源，需特殊防护。

注 3：由于 VDD 串接了 6.8Ω 电阻，此时 I/O 高电平驱动能力将变弱，因此当需要驱动较大负载时，需要在相应 I/O 加驱动电路。

注 4：VREFP 和 VREFN 管脚在复用为 ADC 外部参考时才需要外接电容。

注 5：2.0V ≤ VREF ≤ VDD - 0.1V，且驱动电流不小于 50μA。

注 6：MRST 引脚采用 RC 复位，其中 47KΩ ≤ R1 ≤ 100KΩ，电容 C1 = 100nF，R2 为限流电阻，0.1KΩ ≤ R2 ≤ 1KΩ。

4.2.2 LQFP64 封装芯片最小系统电路

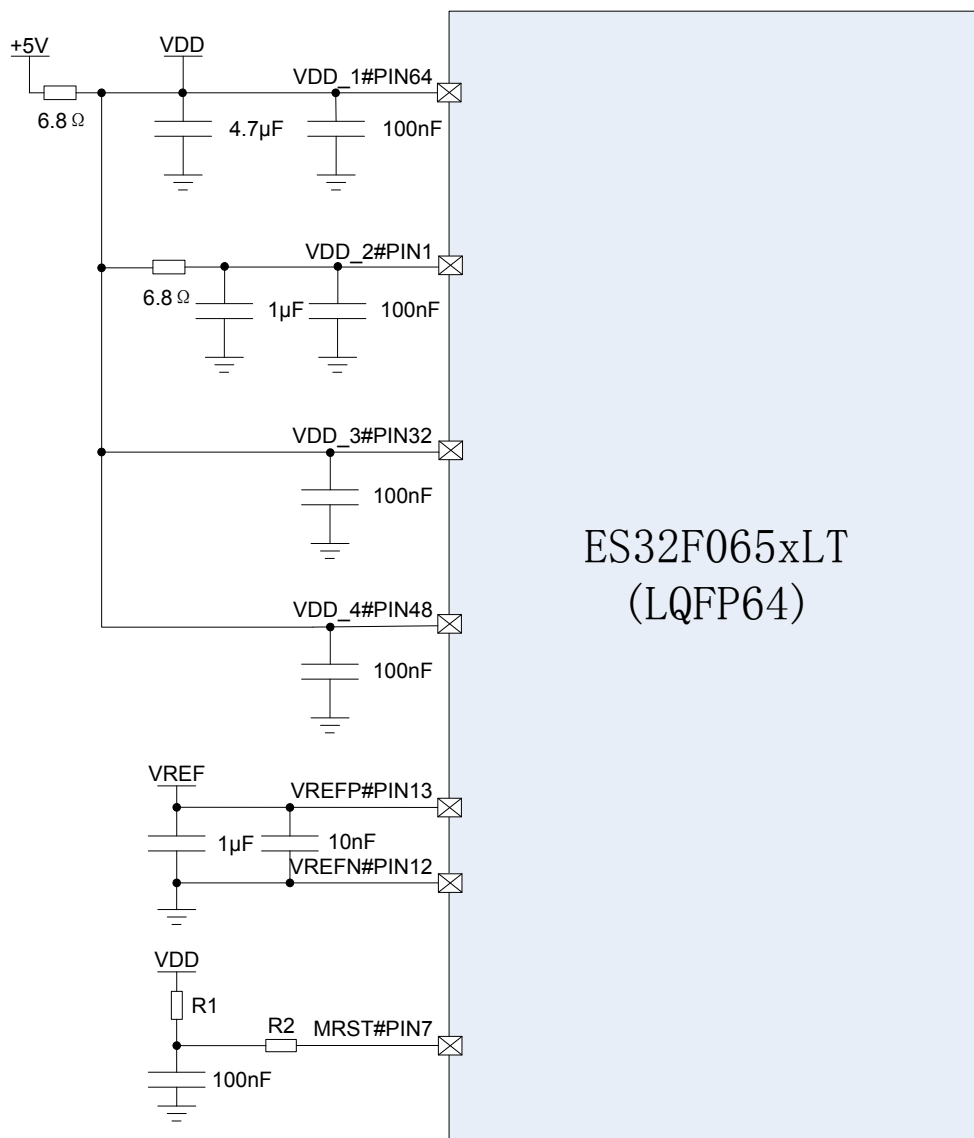


图 4-5 LQFP64 封装芯片电源串接电阻示意图

注 1: 每一组电源必须连接如图所示的陶瓷耦合电容。这些电容必须尽可能地靠近芯片的相应管脚, 才能保证芯片的运行性能。

注 2: PIN64 是芯片主电源, PIN1 是备份域电源, 需特殊防护。

注 3: 由于 VDD 串接了 6.8Ω 电阻, 此时 I/O 高电平驱动能力将变弱, 因此当需要驱动较大负载时, 需要在相应 I/O 加驱动电路。

注 4: VREFP 和 VREFN 管脚在复用为 ADC 外部参考时才需要外接电容。

注 5: $2.0V \leq VREF \leq VDD - 0.1V$, 且驱动电流不小于 50μA。

注 6: MRST 引脚采用 RC 复位, 其中 $47K\Omega \leq R1 \leq 100K\Omega$, 电容 $C1 = 100nF$, $R2$ 为限流电阻, $0.1K\Omega \leq R2 \leq 1K\Omega$ 。

4.2.3 LQFP48/QFN48 封装芯片最小系统电路

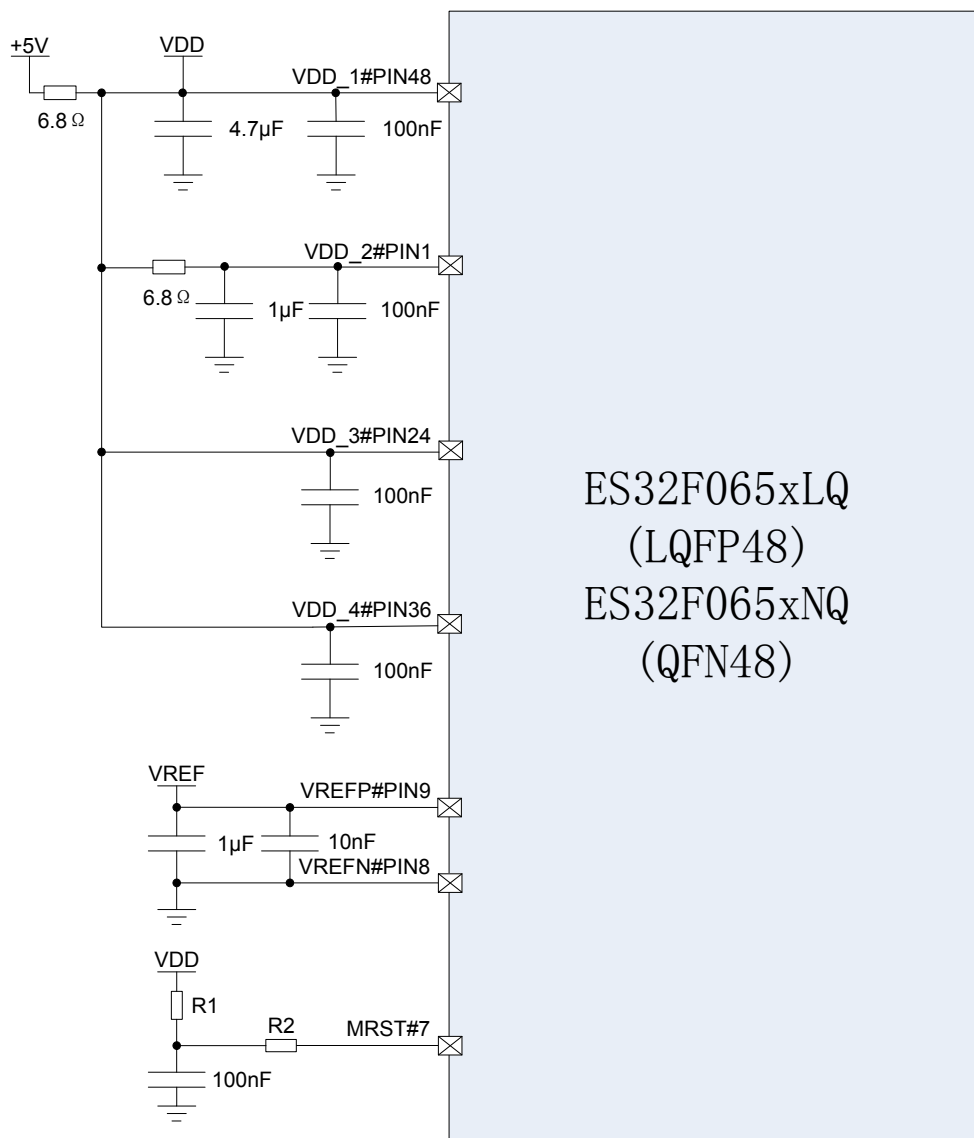


图 4-6 LQFP48/QFN48 封装芯片电源串接电阻示意图

注 1: 每一组电源必须连接如图所示的陶瓷耦合电容。这些电容必须尽可能地靠近芯片的相应管脚, 才能保证芯片的运行性能。

注 2: PIN48 是芯片主电源, PIN1 是备份域电源, 需特殊防护。

注 3: 由于 VDD 串接了 6.8Ω 电阻, 此时 I/O 高电平驱动能力将变弱, 因此当需要驱动较大负载时, 需要在相应 I/O 加驱动电路。

注 4: VREFP 和 VREFN 管脚在复用为 ADC 外部参考时才需要外接电容。

注 5: $2.0V \leq VREF \leq VDD - 0.1V$, 且驱动电流不小于 50μA。

注 6: MRST 引脚采用 RC 复位, 其中 $47K\Omega \leq R1 \leq 100K\Omega$, 电容 $C1 = 100nF$, $R2$ 为限流电阻, $0.1K\Omega \leq R2 \leq 1K\Omega$ 。