



勘误表 MAXQ61H

修订版 B1 勘误表

下面的勘误表列出了 MAXQ61H 修订版 B1 器件的性能与预期值或数据资料描述不一致的内容。Maxim Integrated Products, Inc. 将在重新设计该产品时纠正这些问题。

本勘误表仅适用于 MAXQ61H 修订版 B1 器件。修订版 B1 器件在封装顶部印有一个 yywwB1 字样的六位代码，其中 yy 和 ww 两位分别表示器件的生产年份和周数。如需获得其它 MAXQ61H 裸片修订版的勘误表，请访问我们的网站 china.maxim-ic.com/errata。

1) 在闪存字地址 0x7FFF 不能执行返回操作

说明:

在字地址为 0x7FFF 的闪存存储器执行返回指令时不能返回正确的函数调用结果。

解决途径:

确认该地址不会用于返回指令。

2) V_{DD} 上升较慢或稳定在 V_{POR} 和 V_{RST} 之间时， I_{DD} 可能超出指标

说明:

如果 V_{DD} 电压停留在 V_{POR} 和 V_{RST} 之间， I_{DD} 可能超出规定的最大值，达到 $500\mu A$ 。只有当 V_{DD} 从低于 V_{POR} 开始启动，随后上升到 V_{POR} 以上，但却停留在 V_{RST} 以下时才会发生这种情况。一次标准的掉电过程不会发生这种状况。

解决途径:

确保 V_{DD} 上升速度高于 $1.5V/s$ ，以尽可能缩短停留在 V_{RST} 和 V_{PFW} 之间的时间。系统开发人员须确保低压条件(例如电池放电即将耗尽时)下， V_{DD} 不会在上电过程中停留在 V_{RST} 和 V_{PFW} 之间。

MAXQ61H

修订版 B1 勘误表

3) 进入待机模式前一周期的中断会影响中断优先级的处理

说明:

如果在进入待机模式的前一个周期收到中断，无论原来的优先级如何，该中断将按照较高的优先级处理。如果该中断最初并不具备较高的优先级，错误的中断逻辑将会认为正在处理另外一个具有相同优先级的中断。这种中断故障会妨碍当前同等优先级或较低优先级的有效中断的处理。

解决途径:

可以通过几个途径解决这一问题，在第 4 个问题的解决途径部分给出了一个例程，该软件用于实施解决问题 3 的第 2 个途径和解决问题 4 的第 2 个途径。

- 1) 进入待机模式之前禁止低优先级和中间优先级的中断，退出待机模式后可根据需要立即使能这些中断。如果中断服务程序中需要低优先级或中间优先级的中断，可以在中断服务程序起始部分使能这些中断。
- 2) 退出待机模式时立即执行 NOP 操作，执行一次 POPI 和匹配 PUSH 指令，以清除中断逻辑，删除错误的中断。

4) 进入待机模式前一周期的中断会影响堆栈的完整性

说明:

如果在进入待机模式的前一个周期收到中断，除了中断逻辑推入堆栈的地址外，待机模式逻辑还在堆栈内推入另一返回地址。额外的写堆栈操作使得所有堆栈值偏移一个字。

解决途径:

可以通过几个途径解决这一问题，下面给出了一个例程，用于实施解决问题 3 的第 2 个途径和解决问题 4 的第 2 个途径。

- 1) 进入待机模式之前禁止低优先级和中间优先级的中断，退出待机模式后可根据需要立即使能这些中断。如果中断服务程序中需要低优先级或中间优先级的中断，可以在中断服务程序起始部分使能这些中断。
- 2) 进入待机模式之前把堆栈指针保存在中间位置。完成中断服务程序后，代码将按照激活待机模式的指令恢复。读取堆栈指针并将其与存储值进行比较，如果二者不同，则意味着有额外的数值推入堆栈内。执行 POP 指令，忽略返回值，从堆栈中删除额外的数据，以下程序可以完成这一过程：

MAXQ61H

修订版 B1 勘误表

```
move    CORRECT_STACK_VALUE, SP      ; Save current SP value into a temp register

move    CKCN.4, #1                    ; Enter STOP mode
nop                                           ; NOP that is necessary for errata #2

move    SCRATCH_REGISTER, ACC        ; Save contents of ACC to unused register

repairIPS:
move    ACC, IC                        ; Get register holding IPS bits
and     #0Ch                           ; Mask off everything but IPS bits
cmp     #0Ch                           ; Check for proper value
jump    E, ipsFixed                    ; If IPS == 0x3, no damage was done
popi    ACC                             ; else, do a popi to clear current interrupt level
push    ACC                             ; Need to put the popped value back
jump    repairIPS:                     ; Repeat until IPS == 0x03
ipsFixed:

repairStack:
move    ACC, SP                        ; Get current stack value
cmp     CORRECT_STACK_VALUE           ; Compare against pre-stop value
jump    E, stackFixed                  ; If they match, no damage was done
pop     nul                             ; else, pop off the extra value that was stored
jump    repairStack:                   ; Repeat until SP is back to its original value
stackFixed:

move    ACC, SCRATCH_REGISTER        ; Restore contents of ACC
```

5) 当 MOD[1:0]位由默认值修改为其它数值时，PUSH 指令会导致代码运行错误

说明：

将 MOD[1:0]位(APC[2:0])由默认值修改为其它数值时，在 PUSH 指令后执行读取工作的累加器的任何指令都会导致器件工作不正常。

解决途径：

用户必须确保软件不会修改 AP 寄存器中 MOD[1:0]位的默认值 00b。

6) 使用中断优先级功能会导致错误的中断处理

说明：

在中断后紧跟一个高优先级中断，会导致 CPU 不能正确地响应中断。

解决途径：

不要使用高优先级中断，不要修改 IPR0 和 IPR1 寄存器的复位默认值，应将所有中断源设置为最低(默认)优先级。

MAXQ61H

修订版 B1 勘误表

修订历史

修订号	修订日期	说明	修改页
0	7/10	最初版本	—
1	11/10	增加了勘误信息#5 (PUSH 指令)	3
2	6/11	增加了勘误信息#6 (中断优先级)	3