

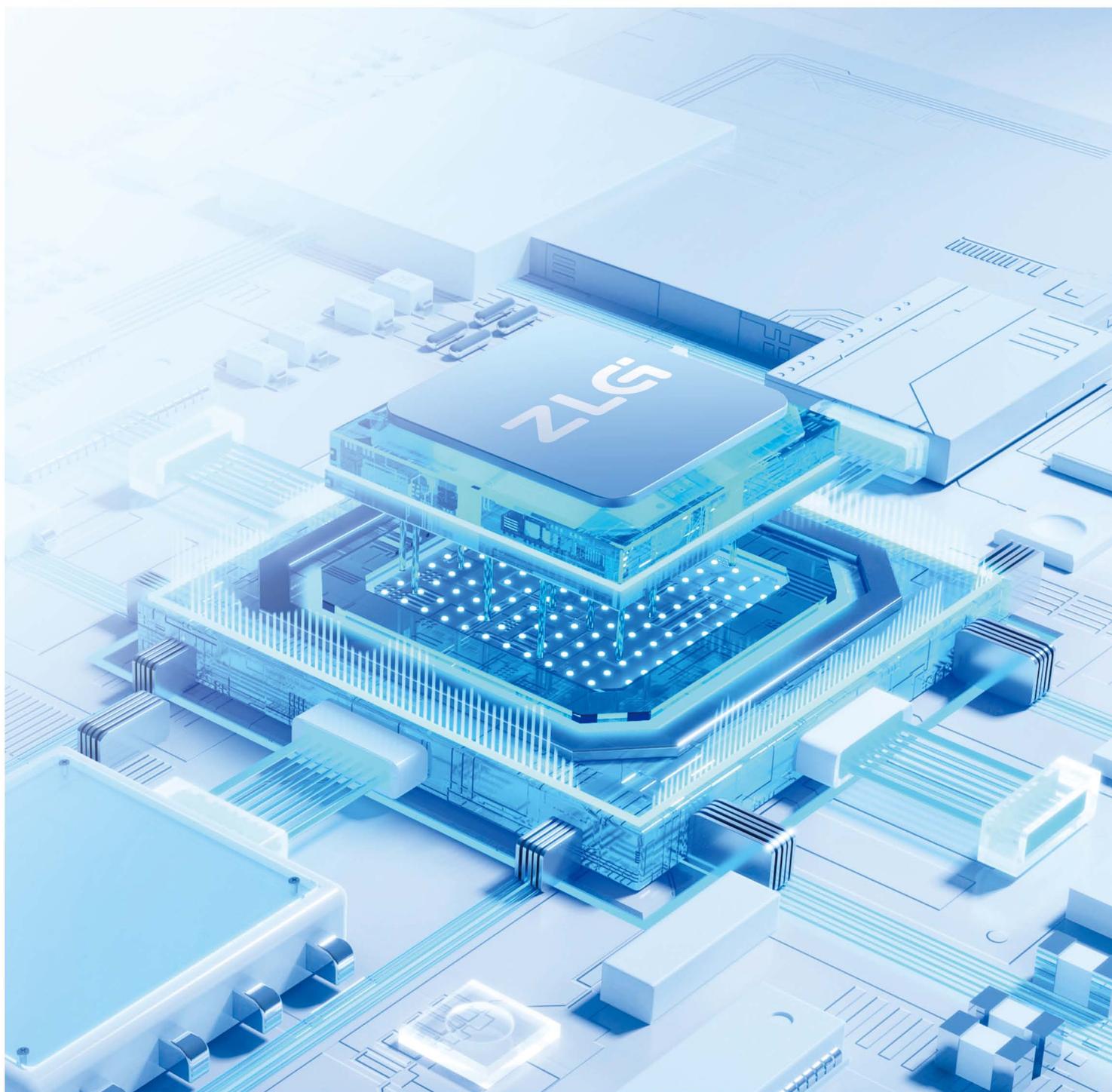
ZLG 致远电子

微文摘

ZLG MICRO DIGEST

2024/12 第12期

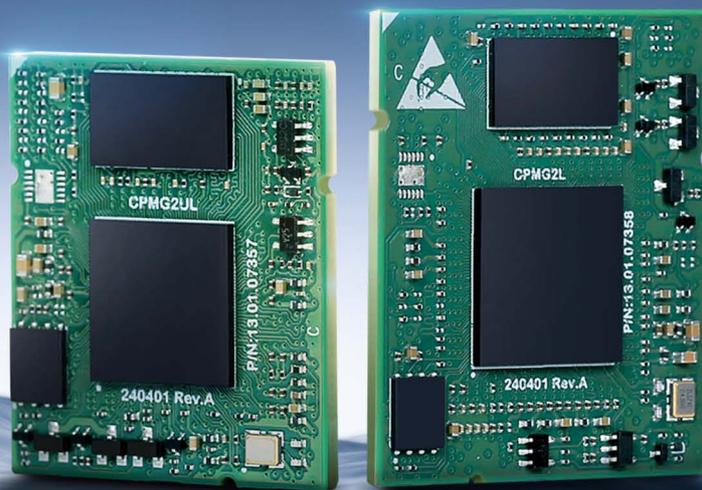
月刊



ZLG 致远电子

CPMG2UL/CPMG2L系列核心板

ZLG重新定义核心板



致远电子官方网站

致远电子官方微信

选型表

型号	CPMG2UL-512M	CPMG2UL-1G
处理器	RZ/G2UL	RZ/G2UL
内核	单核 Cortex®-A55 + 单核 Cortex®-M33	单核 Cortex®-A55 + 单核 Cortex®-M33
主频	1.0 GHz	1.0 GHz
内存	512MB	1GB
操作系统	Linux	Linux
UART	7 路 (含 1 路调试串口)	7 路 (含 1 路调试串口)
CAN FD	2 路	2 路
以太网	2 路千兆	2 路千兆
USB	2 路 USB2.0	2 路 USB2.0
RGB LCD	1 路 24bit	1 路 24bit
MIPI CSI-2	1 路 4lane	1 路 4lane
IIC	3 路	3 路
SPI	3 路	3 路
IIS	3 路	3 路
ADC	2 路 12bit	2 路 12bit
PWM	23 路	23 路
GPIO	79 路	79 路
SDIO	2 路	2 路
JTAG	1 路	1 路
RTC 时钟	支持	支持
独立硬件看门狗	支持	支持
供电电压	3.3V & 2.5V & 1.8V & 1.2V & 1.1V	3.3V & 2.5V & 1.8V & 1.2V & 1.1V
机械尺寸	30mm×34mm	30mm×34mm
环境测试	-40°C ~85°C	-40°C ~85°C
评估套件	ZSOM-EVB+ZSOM-S5501I (含 CPMG2UL-1G 核心板)	ZSOM-EVB+ZSOM-S5501I (含 CPMG2UL-1G 核心板)

CONTENTS

目录

技术平台

EsDA 平台

AWTK-WEB 快速入门 (2)JS 应用程序····· 04

边缘计算

核心板

ZLG 嵌入式笔记 (连载 01) | 非常容易被忽略的电流倒灌····· 06

ZLG 嵌入式笔记 (连载 02) | 电流倒灌揭秘: IO 口损坏与系统故障的真相····· 07

ZLG 嵌入式笔记 (连载 03) | 硬件设计必读: 电流倒灌解决方案····· 08

ZLG 嵌入式笔记 (连载 04) | 电源设计避坑 (上)····· 10

ZLG 嵌入式笔记 (连载 05) | 电源设计避坑 (下)····· 12

ZLG 嵌入式笔记 (连载 06) | Linux 的 RTC 回到了 1970 年, 是时光倒流的么?····· 14

ZLG 嵌入式笔记 (连载 07) | RTC 时钟偶发性延时或超时该怎么办?····· 16

ZLG 嵌入式笔记 (连载 08) | 那些不能随便添加的元器件····· 18

ZLG 嵌入式笔记 (连载 09) | 电平匹配问题, 简单却容····· 20

ZLG 嵌入式笔记 (连载 10) | 为什么你的网卡工作会不正常? (上)····· 23

ZLG 嵌入式笔记 (连载 11) | 为什么你的网卡工作会不正常? (中)····· 25

ZLG 嵌入式笔记 (连载 12) | 为什么你的网卡工作会不正常? (下)····· 26

ZLG 嵌入式笔记 (连载 13) | 以太网带宽总是跑不满, 是什么原因?····· 28

ZLG 嵌入式笔记 (连载 14) | IIC 总线上拉电阻阻值如何确定?····· 30

ZLG 嵌入式笔记 (连载 15) | RS485 偏置电阻和终端电阻有什么作用、怎么选取?····· 31

行业控制器

【产品应用】ZMC600E 运动控制器 T 型、S 型速度规划详解····· 33

互联互通

CAN-bus 总线

【CAN 总线知识】为何总线“镰刀”波形频频发生?····· 37

感知控制

电源与隔离

【热电偶测温】揭开热电偶滤波电路的神秘面纱····· 39

AWTK-WEB 快速入门(2) - JS 应用程序

ZLG 致远电子 2024-12-04 11:34:04

AWTK 可以使用相同的技术栈开发各种平台的应用程序。有时我们需要使用 Web 界面与设备进行交互，本文介绍一下如何使用 JS 语言开发 AWTK-WEB 应用程序。

用 AWTK Designer 新建一个应用程序

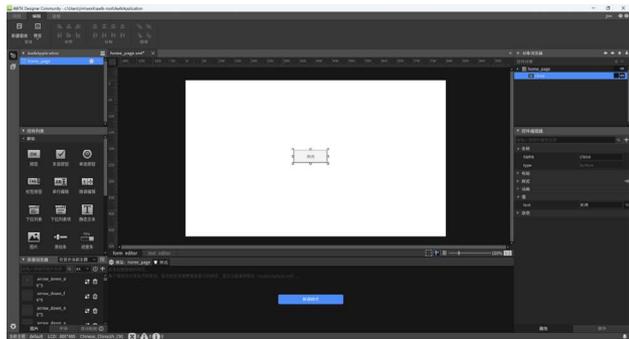
先安装 AWTK Designer:

<https://awtk.zlg.cn/web/index.html>

1. 新建应用程序

这里假设应用程序的名称为 AwtkApplicationJS，后面会用到，如果使用其它名称，后面要做相应修改。

- 在窗口上放置一个按钮；
- 将按钮的名称改为 “close”；
- 将按钮的文本改为 “关闭”。



2. 为按钮编写代码

2.1 删除 src 目录下全部文件（留着也可以，只是看起来比较乱），在 src 目录创建 js 目录。

2.2 在 src/js 下创建 application.js，内容如下：

```
function applicationInit() {
  home_page_open();
}
```

applicationInit()

2.2 在 src/js 下创建 application.js，内容如下：

```
function applicationInit() {
  home_page_open();
}
```

applicationInit()

applicationInit()

2.3 在 src/js 下创建 home_page.js，内容如下：

```
function home_page_open() {
  var win = TWindow.open("home_page");
  var close = win.lookup("close", true);

  close.on(TEventType.CLICK, function(evt) {
    var e = TPointerEvent.cast(evt);
    var target = TButton.cast(e.target);

    console.log("click at " + e.x + " " + e.y + " on " + target.name);
    console.log("global: " + TGlobal.getPointerX() + " " + TGlobal.getPointerY() + " " + TGlobal.isPointerPressed());
    return TRet.OK;
  });

  win.layout();
}
```

注意：控件的名称一定要和 home_page.xml 保持一致。

3. 在 AWTK Designer 中，执行“打包”“编译”“模拟运行”



正常情况下可以看到如下界面：



点击“关闭”按钮，退出应用程序。

编写配置文件

具体格式请参考，特殊平台编译配置：

https://github.com/zlgopen/awtk/blob/master/docs/build_config.md

md

这里给出一个例子，可以在此基础上进行修改，该文件位于：
examples/AwtkApplicationJS/build.json

```
{
  "name": "AwtkApplicationJS",
  "version": "1.0",
  "app_type": "js",
  "copyright": "Guangzhou ZHIYUAN Electronics Co.,Ltd.",
  "themes": ["default"],
  "sources": [
    "src/js/*.js"
  ]
}
```

编译 WEB 应用程序

进入 awtk-web 目录，不同平台使用不同的脚本编译：

Windows 平台

```
./build_win32.sh examples/AwtkApplicationJS/build.json release
```

Linux 平台

```
./build_linux.sh examples/AwtkApplicationJS/build.json release
```

MacOS 平台

```
./build_mac.sh examples/AwtkApplicationJS/build.json release
```

请根据应用程序所在目录，修改配置文件的路径。

运行

1. 正常启动

```
./start_web.sh
```

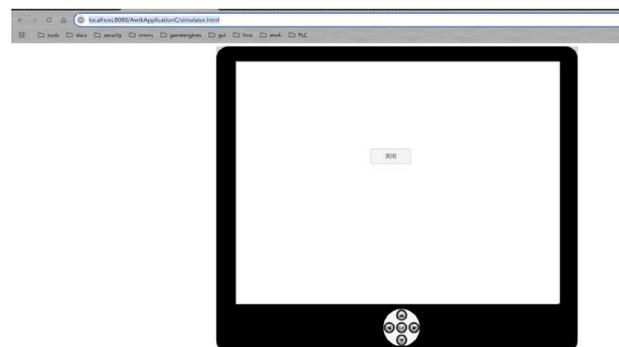
2. 调试启动

```
start_web_debug.sh
```

3. 用浏览器打开 URL：http://localhost:8080/AwtkApplicationJS，可以看到：



4. 用浏览器打开 <http://localhost:8080/AwtkApplicationJS/simulator.html>，可以看到：



5. 用浏览器打开 <http://localhost:8080/AwtkApplicationJS/app.html>，可以看到：



AWTK-WEB-JS 和 AWTK-JS 不同

两者在 API 上基本上是一样的，但是由于浏览器和 Jerryscript 的差异，导致少量功能无法做到完全一致，需要应用程序自行规避。

1. AWTK-WEB-JS 不支持模态对话框。

2. AWTK-WEB-JS 中的 print 函数是打印。请使用 console.log 输出调试信息。

3. AWTK-WEB-JS 不支持自动回收在 wasm 中分配的对象（如：color/rect/value 等）。如果希望两者通用，请避免使用这类函数。



图像显示应用芯片 ZMP110X

[点击购买](#)

【ZLG嵌入式笔记(连载01) | 非常容易被忽略的电流倒灌】

ZLG 致远电子 2024-12-09 11:34:47

EsDA 是一套面向工业智能物联领域的嵌入式系统设计自动化工具集，包含实时操作系统 AWorksLP、低代码开发平台 AWStudio、资源管理平台 AXPI、跨平台 GUI 引擎 AWTK 和云服务平台 ZWS，旨在提高嵌入式软件开发的效率、性能和可扩展性。

电流倒灌问题在 5V 电平的单片机时代几乎不会发生，主要是因为 5V 单片的 IO 耐压值高，单片机内部结构对 IO 保护设计很好。到了 3.3V 单片机时代，这类问题有一定的偶发性，但还是很少，尽管单片机 IO 口耐压值有所降低，但内部保护设计依然很到位，同时单片机上电后到程序运行的时间很短，IO 口正确初始化后，能很好的与外设电平进行匹配，也能避免电流倒灌。而到了现在的 MPU，处理器规模都比较大，IO 口很多，系统电源通常有好几路不同的电压，而且还有严格的上电时序要求，再加上处理器的多级启动，启动时间较长，从上电到完成 IO 口初始化这段时间，远远比以前的单片机长，这段时间如果不谨慎处理电平，就有可能产生电流倒灌。

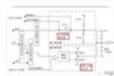
电流倒灌是一个非常隐蔽的问题，属于“慢性病”，在产品使用中很难被立即发现，只有在应用现场长时间运行后才有可能出现问题，出现的问题非常多样化，表现各异，就算给出解决办法，效果也不是立竿见影，而是需要通过长时间运行才能验证改善方案的有效性。

如果一个产品在现场运行较长一段时间后出现了故障，而用同一型号产品在实验室却无法复现，在基本定位为硬件问题的情况下，可以考虑排查电路设计上有无存在电流倒灌。结合多年来的应用经验，如下故障 / 问题可以往电流倒灌方面进行排查：

1. 产品在现场运行几个月后出现偶发性重启或者宕机，更换核心板后故障消失；
2. 产品在现场运行几个月后，网络偶发性异常，或者 IO 口控制设备异常，更换核心板后恢复正常；
3. 产品在现场运行几个月后，系统开不了机，更换核心板后正常，故障板在实验室更换底板后依然无法启动系统。

电流倒灌却又是一个很常见的问题，在网络上以“IO 电流倒灌”为关键词进行搜索，可以找到非常多的相关案例和解决办法的文章，如图 1 所示。

[逻辑电平案例:从IO口结构分析电流倒灌的原因及解决措施-电...](#)



STM32的IO口结构 2、电流倒灌的原因及解决措施 当使用CMOS型器件作为接口芯片时,如果Vcc2断电,Vcc1继续供给G1,G1的高电平输出电流将通过D1向Vcc2上的电容充电(该充电电流将使D1迅速过载并使其损坏...

电子发烧友

[MSP430IO口拉电流灌电流问题 - 百度文库](#)

1页 发布时间: 2023年09月02日

灌电流就是电流流向IC元件,拉电流就是电流从IC元件流出。不管是灌电流还是拉电流,每个IO端口的输出晶体管都能够限制输出的电流最大为6mA,保证系统安全。Rmin=V/Imax=3.3V/6...

百度文库

[GPIO电流倒灌问题\(防倒灌\)_io防倒灌-CSDN博客](#)

在实际应用中,当串口芯片与主控IC譬如MCU等器件一同使用时,如果串口直连的双方器件有一方不需要供电工作时,要注意一方电流对另一方电流的倒灌导致未供电的芯片继续工作的情况...

CSDN博客

[倒灌电流后果_单片机 - 电子发烧友网](#)

2023年11月27日 如何解决GPIO电流倒灌问题 在单片机的使用中,有这样一种现象:当断开单片机的

图1 网络上的“IO 电流倒灌”搜索结果

电流倒灌和IO电流倒灌

1. 电流倒灌

电流倒灌是指电流方向与常规流动方向相反的现象。在电路中，电流的方向通常是从电源的正极流出，经由负载流入电源的负极。然而，在某些情况下，如电源电压过高、电路保护措施不足或电路设计错误等，电流可能会反向流动，从电源的负极流回正极，这种现象就称为电流倒灌。

2. IO 电流倒灌

IO 电流倒灌是指在电子设备中，电流从电源线或信号线流回芯片或电路的现象。这通常是由于电源或信号线的配置不当，或者是因为电路设计错误导致的。

引脚电平冲突也有可能引起 IO 电流倒灌。例如某处理器引脚默认输出了低电平，而所连接外设的引脚默认输出了高电平，此时就会产生从外设到处理器的 IO 灌电流。

3. 电流倒灌的后果

电流倒灌可能产生的后果，在症状上具有不确定性、多样性，以及问题的隐蔽性，经常让人捉摸不透。轻则可能导致信号失真或不稳定，从而影响设备的性能和稳定性，重则某个或者多个 IO 工作不正常，或者 IO 引脚损坏，引起相应外设控制异常，也有引起某些通信接口异常，再重一点就系统死机，出现莫名其妙的故障，更严重的则烧坏 CPU，只能走返修途径或者报废。

关注我们，下期分析：为什么电流倒灌会出现以上现象？



ZLG创新性CPM核心板

点击购买

ZLG嵌入式笔记(连载02) | 电流倒灌揭秘：IO口损坏与系统故障的真相

ZLG 致远电子 2024-12-11 11:35:48

本期文章将继续深入了解电流倒灌，分析嵌入式系统中 IO 口损坏和系统稳定性问题的根本原因。

在上期的工程笔记中，我们了解了电流倒灌并探讨了电流倒灌可能导致的一系列问题，包括 IO 口损坏、系统死机、系统不开机或休眠及唤醒异常。这些问题虽然听起来颇为技术性，但它们对嵌入式系统的正常运行至关重要。本期，我们将深入分析这些现象背后的原因。

IO口损坏

IO 端口可分为高阻、三态、推挽、开漏等类型，除了功能性区别外，几乎所有 IO 口内部都会存在如图 1 所示结构的四个二极管。

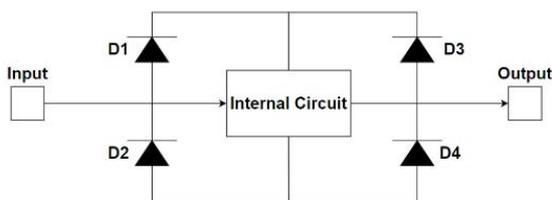
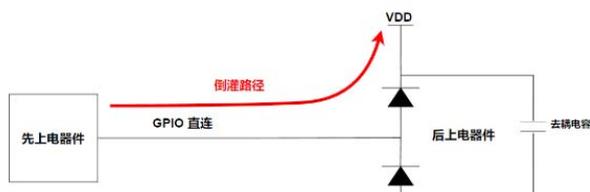


图1 IO口内部结构

- D1 在大多数集成电路中起着防静电功能，同时辅助起输入端限幅作用。
- D2 是半导体集成所产生的寄生二极管，有一定的放电保护功能。
- D3 用于保护 CMOS 电路在放电时的干扰，在大多数双极性器件中也存在此二极管。
- D4 是晶体管的集电极（双极性）或漏极（场效应管）的寄生二极管，有放电作用。

这些等效或者原本就存在的二极管为电流倒灌提供了回路。

下面结合图 2 来分析电流倒灌破坏 IO 口的过程。在一个具有上电顺序的情况下（或右侧器件断电或者拔插动作），左侧的电压如果足够大，那么就会通过二极管向右侧的 VDD 充电，也就是去耦电容的充电，这样就会使二极管急速过载损坏，电容本身也有可能损坏，瞬时的大电流甚至会直接击毁元件本身，再好的结构也会使得器件逻辑工作不正常。



[逻辑电平案例:从IO口结构分析电流倒灌的原因及解决措施-电...](#)



STM32的IO口结构 2、电流倒灌的原因及解决措施 当使用CMOS型器件作为接口芯片时,如果Vcc2断电,Vcc1继续供给G1,G1的高电平输出电流将通过D1向Vcc2上的电容充电(该充电电流将使D1迅速过流并使其损坏...

电子发烧友

[MSP430IO口拉电流灌电流问题 - 百度文库](#)

1页 发布时间: 2023年09月02日

灌电流就是电流流向IC元件,拉电流就是电流从IC 元件流出。不管是灌电流还是拉电流,每个IO端口的输出晶体管都能够限制 输出的电流最大为6mA,保证系统安全。Rmin=V/Imax=3.3V/6...

百度文库

[GPIO电流倒灌问题\(防倒灌\)_io防倒灌-CSDN博客](#)

在实际应用中,当串口芯片与主控IC譬如MCU等器件一同使用时,如果串口直连的双方器件有一方不需要供电工作时,要注意一方电流对另一方电流的倒灌导致未供电的芯片继续工作的情况...

CSDN博客

[倒灌电流后果_单片机 - 电子发烧友网](#)

2023年11月27日 如何解决GPIO电流倒灌问题 在单片机的使用中,有这样一种现象:当断开单片机的

图1 网络上的“IO电流倒灌”搜索结果

电流倒灌和IO电流倒灌

1. 电流倒灌

电流倒灌是指电流方向与常规流动方向相反的现象。在电路中，电流的方向通常是从电源的正极流出，经由负载流入电源的负极。然而，在某些情况下，如电源电压过高、电路保护措施不足或电路设计错误等，电流可能会反向流动，从电源的负极流回正极，这种现象就称为电流倒灌。

2. IO电流倒灌

IO 电流倒灌是指在电子设备中，电流从电源线或信号线流回芯片或电路的现象。这通常是由于电源或信号线的配置不当，或者是因为电路设计错误导致的。

引脚电平冲突也有可能引起 IO 电流倒灌。例如某处理器引脚默认输出低电平，而所连接外设的引脚默认输出了高电平，此时就会产生从外设到处理器的 IO 灌电流。

3. 电流倒灌的后果

电流倒灌可能产生的后果，在症状上具有不确定性、多样性，以及问题的隐蔽性，经常让人捉摸不透。轻则可能导致信号失真或不稳定，从而影响设备的性能和稳定性，重则某个或者多个 IO 工作不正常，或者 IO 引脚损坏，引起相应外设控制异常，也有引起某些通信接口异常，再重一点就系统死机，出现莫名其妙的故障，更严重的则烧坏 CPU，只能走返修途径或者报废。

关注我们，下期分析：为什么电流倒灌会出现以上现象？



ZLG创新性CPM核心板

[点击购买](#)

ZLG嵌入式笔记(连载03) | 硬件设计必读：电流倒灌解决方案

ZLG 致远电子 2024-12-12 11:35:27

通过案例分析，我们将探讨如何通过电路设计规避潜在的电流倒灌风险。本文旨在为您提供实用的设计指导，以提升产品的可靠性和性能。

在电子设备的设计与维护中，IO 电流倒灌是一个需要特别关注的问题。为了防止这一现象的发生，我们可以采取以下措施：
1. 在电源和信号线之间添加适当的保护电路或保护器，以确保电流不会流向错误的方向。
2. 正确地设计和配置电路，也可以减少 IO 电流倒灌的风险。
在电子设备中，使用适当的电阻、电容、二极管等元件来配置电路，以防止电流倒灌。

案例分析

案例1：底板外设先于核心板上电，IO电流倒灌到SOC芯片内部

解决这样的电流倒灌，需要注意研究核心板电源供电电路的资料。以 ZLG 的 M3568 核心板为例，核心板提供了一个 PWR_OK 引脚，核心板上电完成后，该引脚输出高电平。用 PWR_OK 引脚来控制使能二级电源，则可以规避 IO 电流倒灌问题。

图 1 为电源系统二级电源，使用 SOC 配套的 PMU 输出的 PWR_OK 来使能二级电源，符合上电时序要求。M3568 核心板完成启动配置时间大概为 200ms，PWR_OK 也大概是在 5V 供电后的 200ms 会起来，可以保证核心板上电完毕后底板其他电源再上电。

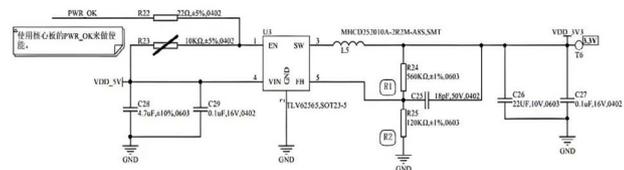


图1 二级电源

案例2：底板控制电路或芯片通过外设上拉电路拉高核心板引脚，上拉电源域比SOC对应的GPIO的供电电源域先上电，产生IO电流倒灌

如图 2 所示，SOC 的 GPIO 通过上拉电阻上拉到底板电源域，如果底板电源域比 SOC 对应的 GPIO 的供电电源域先上电，则会发生倒灌，长期可能引起 SOC 工作异常，甚至可能进一步导致底板电源无法关闭。所以在需要使用按键控制设备开关机时，必须严格控制底板电源的关断控制。

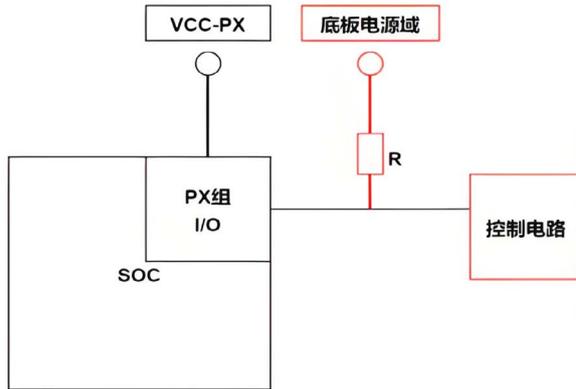


图2 底板电源和SOC

解决这样的问题，可参考 Coral3568 核心板的底板电源电路设计。该核心板带 PWRON 和 PWROK 两个引脚，可支持开关机按键电源设计（短按开机，长按关机）。

电路如图 3 所示，将 PWRON 信号和 PWROK 信号通过二极管相连，PWROK 信号连正极，PWRON 信号连负极，这样按键按下时 PWRON 信号可以拉低 PWROK，关闭底板电源。另外通过该按键关机后，核心板 5V 需保持上电状态，如果需要关闭底板的 5V 外设电源，则需将上述两者分为两个电源域。

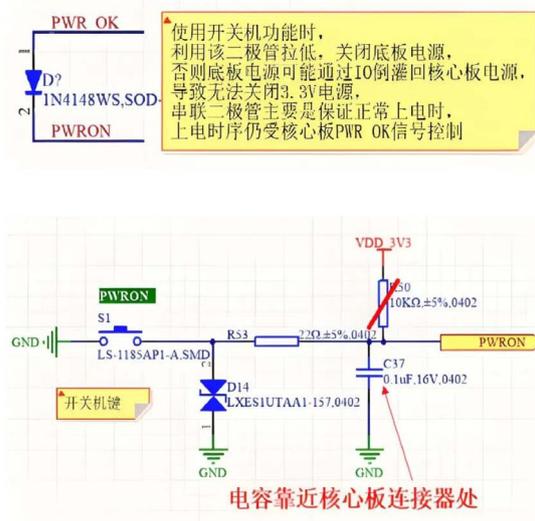


图3 带开关机电源设计

案例3：接口电路设计不合理产生IO电流倒灌

在电路设计中，会遇到处理器 IO 电平与模块 IO 的电平不相同的问题。为保证正常通信，需要进行电平转换。若直接相连，像 TTL 电平就会出现电流倒灌现象。电平转换要求：

- $VOH \geq VIH$, $VOL \leq VIL$;
- 对于多电源系统，某些器件不允许输入电平超过电源电压，针对有类似要求的芯片，注意电路做适当保护；
- 电平转换影响通信速度。

HDMI 的 IIC 采用的是 5V 电平标准，而通常板载的 I2C 是 3.3V 或 1.8V 电平标准。M3568 评估底板 HDMI 设计 NMOS 电平转换，如图 4 所示。

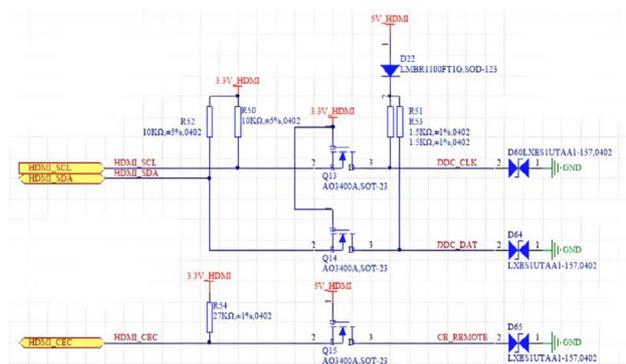


图4 M3568评估底板HDMI设计NMOS电平转换电路

- 左侧板载输入高电平，UGS=0V，NMOS管截止，右侧输出为5V高电平；
- 左侧板载输入低电平，UGS=3.3V，NMOS管导通，右侧输出为板载低电平；
- 右侧接口输入高电平，UGS=0V，NMOS管截止，左侧输出为3.3V高电平；
- 右侧接口输入低电平，NMOS体二极管导通，UGS=2.6V（体二极管有压降），只要NMOS管开启电压小于UGS，NMOS管导通，左侧输出为接口低电平。

此外，HDMI 接口供电电源为底板 5V 电源域，在如图 5 所示的电路中，加肖特基二极管 D15 是为了防止在底板掉电后出现电流倒灌。

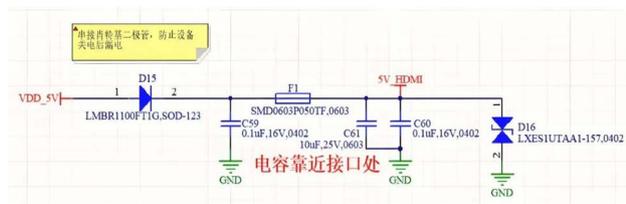


图5 HDMI 5V电源防倒灌处理

总结

采用致远电子的核心板开发产品，也需要考虑底板的上电时序，但无需像核心板内处理器上电时序那么复杂，只要规避发生电流倒灌即可。

针对硬件系统中可能存在容易发生电流倒灌问题，在进行硬件电路设计时，需遵循以下原则：

- 若常供电部分比 PMU&SOC 优先上电或 PMU&SOC 掉电后常供电部分仍有供电。为避免发生倒灌，此时 PMU&SOC 相连接的通讯信号或控制信号都需做电平隔离处理。
- 任何常供电电源不可通过上拉电阻连接到 PMU&SOC。
- SOC 各组 GPIO 上拉电阻需选择上拉至其对应供电电源域。
- 外设芯片供电电源由 MCU 或 SOC 控制上电时序。
- 外设芯片不能设计成硬件上电使能或复位，避免 SOC 漏电至外设芯片。



ZLG嵌入式笔记(连载04) | 电源设计避坑 (上)

ZLG 致远电子 2024-12-13 11:34:36

产品上量后,通常都会有降成需求。多年来,接触过不少产品降成案例,在电源上下刀过猛,引发了产品偶发性问题,带来了很不好的负面影响。本文将对这些案例进行总结,提供电源设计参考,确保产品降成不降质量。

产品降成是商业活动中的正常行为,可以在产品设计、研发、生产、运输、销售、维护等各个环节进行考虑。对于研发工程师而言,在降成方面通常是从设计入手,选择什么样的方案能节省物料成本、研发成本或者生产成本。这也是要看多方面的综合成本,如果选择了一个方案,单从硬件 BOM 成本看是降低了,但如果带来了软件研发成本的急剧增加,或者生产和维护成本大幅增加,那这个方案也是不合适的。

至关重要的一点,产品降成一定不能降低对产品的品质要求,至少要确保能满足产品设计参数要求,以及能通过相关测试。所以降成不能以牺牲产品品质为代价,否则带来的维护成本会大幅增加,严重影响公司口碑。

在一些产品或系统中,电源成本占比很高,所以就有人想在电源上打主意。遇到过不少案例,在电源设计上做了物料成本节省,在电源最大功率、电源质量等方面打了折扣,带来了极大的隐患。电源系统为系统或产品工作提供能源,电源质量的好坏直接影响着产品的稳定性,稳定可靠的电源是产品和系统稳定运行的基础,如果电源不稳定,系统的可靠性就无从谈起,千万不要在电源上抠成本。

下面总结了电源设计方面的一些考量因素,仅供参考。

电源功率

如果电源功率不足以满足产品的需求,会导致产品无法正常工作或出现故障。例如,如果一个电源额定功率是 500W,峰值功率是 600W,配套给正常工作就需要 600W 功率的产品供电,有可能导致产品供电不足引起异常,且电源长时间处于高负荷状态,也会加速其老化,缩短使用寿命。所以在选择系统电源时,一定要注意额定功率和峰值功率,不能把峰值功率当正常功率使用。

使用电源芯片设计供电电路时,也一定要有足够的设计余量,不能把电源芯片输出的最大值当成设计值来使用,而应当将设计值控制在典型值以内或者附近,当然更不能超出芯片本身的最大值。例如芯片输出电流典型值 1.0A,最大值为 1.50A,就一定不能接超过 1.5A 的负载,而应当控制在 1.0A 附近。

实际案例:

某智能家居系统在产品推向市场后,用户反馈设备经常出现重启现象。经检查和分析,发现电源的输出功率不足以支持设备的正常运行,导致设备因过载而重启。

应对措施:

在选择电源时,应充分考虑产品的功耗需求。对于需持续稳定运行的设备,应选择输出功率足够大的电源,以确保为产品提供稳定的电力支持。在设备设计过程中,还应考虑电源的散热设计和负载计算,避免因电源过载而引起的设备故障。

电源稳定性

电源稳定性差会导致电压和电流波动,从而影响产品的性能和稳定性。如果电源的输出电压或电流不稳定,可能会导致产品出现重启、死机、数据丢失等问题。此外,如果电源的响应速度慢,不能及时响应负载变化,也会影响产品的性能和稳定性。

实际案例:

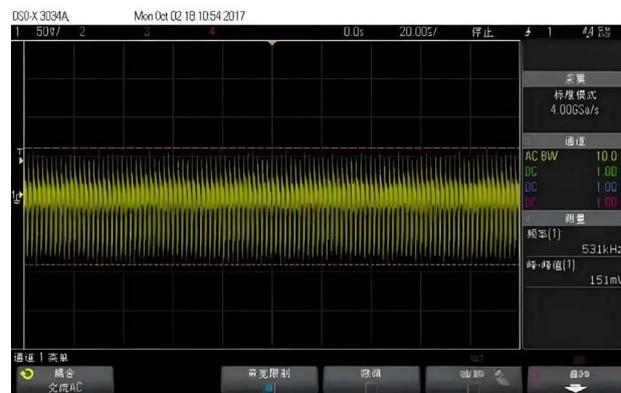
计算机在充电时突然断电,导致计算机损坏。经调查发现,电源在充电过程中的输出电压波动较大,导致计算机电路板损坏。

应对措施:

选择输出电压稳定性好的电源,以减少对产品的干扰和损坏。对于对电压波动敏感的设备,如计算机、手机等,应特别关注电源的稳定性。此外,还可优化稳压电路来提高设备的抗电压波动能力。

纹波控制

纹波控制是衡量电源性能的重要指标之一。良好的纹波控制可减少电源输出中的交流成分,提高产品的稳定性和可靠性。如果纹波控制不良,可能会导致产品出现噪音、干扰等问题,甚至损坏。因此,在选择电源时,应关注其纹波控制能力,选择具有优良纹波性能的电源。



(图片来源于网络,侵权)

实际案例:

某显示器在显示图像时出现明显的噪点和干扰。经过维修和检查,发现电源的纹波控制不良,导致显示器出现噪点和干扰问题。

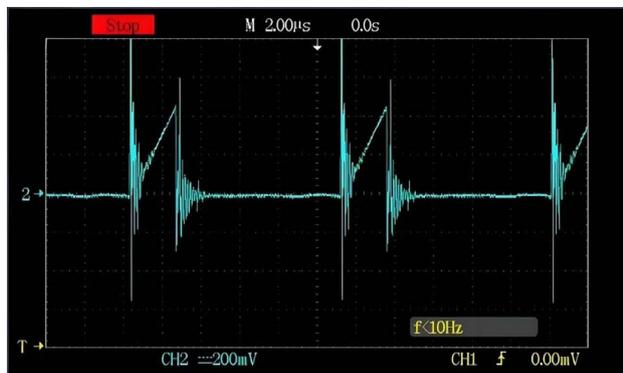
应对措施:

选择具有优良纹波性能的电源,以减少对显示器的干扰和损坏。对于

对图像稳定性要求高的设备,如显示器、电视等,应特别关注电源的纹波性能。同时,可优化滤波电路来降低纹波对设备的影响,提高设备的图像质量和稳定性。

尖峰和浪涌

尖峰和浪涌是电源中比较罕见的异常现象。尖峰是指电压或电流突然出现高幅度峰值,通常持续时间较短。浪涌是指电压或电流突然出现大幅度波动,通常持续时间较长。如果电源中出现尖峰或浪涌,可能会导致产品损坏或出现其他不可预测的问题。



(图片来源于网络,侵权)

实际案例:

某工业生产线上传感器的雷电天气下经常出现故障。经过调查和分析,发现电源在雷电天气下出现了尖峰电流和浪涌现象,导致传感器线路过热而出现故障。

应对措施:

选择具有防尖峰和浪涌功能的电源,以保护产品免受电流 / 电压冲击的损害。对于大电流设备,如工业生产线上传感器、电机等,应特别关注电源的尖峰和浪涌保护能力。同时,可采取相应的防护电路和保护措施来降低电流 / 电压冲击对设备的影响。

下一期我们将继续深入探讨电源设计的其他考虑因素,包括启动电流和冲击电流、备用电源、电源设计余量和其他注意事项。



6W隔离宽电压输入电源模块

[点击购买](#)

ZLG嵌入式笔记(连载05) | 电源设计避坑 (下)

ZLG 致远电子 2024-12-16 11:35:02

产品上量后,通常都会有降成需求。多年来,接触过不少产品降成案例,在电源上下刀过猛,引发了产品偶发性问题,带来了很不好的负面影响。本文将对这些案例进行总结,提供电源设计参考,确保产品降成不降质量。

在上期《电源设计避坑(上)》中,我们讨论了电源设计中的电源功率、稳定性、纹波控制以及尖峰和浪涌的问题,并结合实际案例提出了相应的应对措施。接下来,我们将继续探索电源设计的其他关键考量因素。

启动电流和冲击电流

电机在启动时会产生较大的启动电流和冲击电流,这可能会对电源造成压力和不稳定。如果电源没有足够的承载能力,可能会导致电源过载、电压波动等问题,从而影响产品的性能和稳定性。因此,在为电机等大功率设备选择电源时,需充分考虑其启动电流和冲击电流的大小,选择合适的电源规格和保护措施。

当电机还未启动转速为零时,加上额定 380V 电压而启动,它瞬间的线电流称为启动电流,这个启动电流是从小至大而又回到额定电流。异步电机直接启动时,其启动电流最大可达到额定电流的 4~7 倍,这是对电网造成影响的主要因素。

首先是使电网电压瞬间下降,俗称的压降。特别在电源容量小(电力变压器容量小)和大功率电机启动的情况下,电压下降更大,不仅使该电机启动困难,还影响到电源线路上其他电机的正常运行。

电机启动时的瞬时大电流,对电机本身和电网电源都有非常大的影响。另一方面,过大的启动电流将使电机和线路上的电能损耗增加,特别是在频繁启动、启动较慢、或启动过程较长的情况下,电能损耗更大,更容易造成电机严重发热。所以,只要电机在启动时对供电线路有压降影响的都应限制其启动电流。

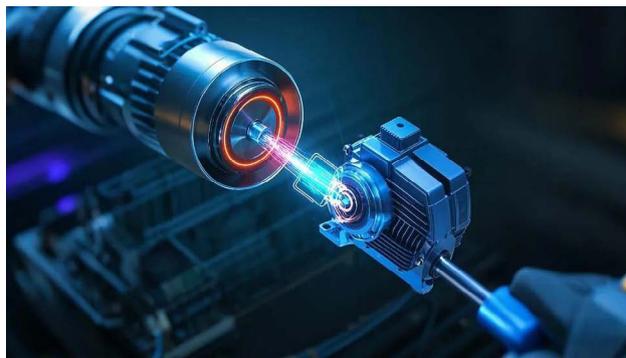
启动电流是指电气设备(感性负载)在通电的瞬间到运行平稳的短暂的时间内的电流变化量,这个电流一般是额定电流的 4~7 倍。国家规定,为了线路的运行安全及其它电气设备的正常运行,大功率的发动机必须加装启动设备,以降低启动电流。

在电路学中,给负载通电的一瞬间,通常会产生大电流,这就是冲击电流。这个现象主要体现在容性负载中,例如电容,在上电一瞬间是相当于短路的,瞬间电流理论上无限大的。但电机(直流和三相)作为感性负载,如果表现其固有机械特性,怎么会有很大的启动电流呢?

我们知道,电感具有阻止电流变化的作用,能够瞬间承受较大的电压,因此感性负载是有助于稳定电流的。电机是一种感性负载,但在电机启动的一瞬间,由于电机定子和转子之间相对运动的速度几乎为 0,即没有切割

场的运动,就不会在电路中产生反电动势(互感电压为 0),忽略线圈自感的作用。此时,几乎所有的电压都加在了电路的电阻上,由于电阻很小,因此电流很大。这就是说,并不是因为电机是感性负载而导致大的冲击电流,而是因为缺少切割磁场的运动,没有互感电动势造成的。

这类电源建议采取软启动设计,在电源启动过程中逐渐改变机械特性,调整电路参数,使启动电流逐渐增加到正常值的一种方法,可以避免形成较大的冲击电流。常见的交流电机的启动方法有直接启动、串电阻启动、自耦变压器启动、星三角减压启动及软启动器和变频器启动的方法来减小对电网的影响。



实际案例:

某工业机器人生产线在启动时,机器人经常出现启动缓慢或无法启动的问题。经过调查和分析,发现电源在机器人启动时无法提供足够的启动电流,导致机器人无法正常启动。

应对措施:

选择具有足够承载能力的电源,以应对电机等大功率设备的启动电流和冲击电流。在选择电源时,应充分考虑设备的启动电流和冲击电流的大小,选择合适的电源规格和保护措施。此外,还可采取软启动电路等辅助措施来降低启动电流对电源的影响。

备用电源

在重要场合,如医院、数据中心等,需保证电源的稳定性和不间断供电。因此,他们会使用备用电源来应对突发情况,如电源故障或电力中断。如果在这些场合中使用了降成的电源方案,可能会导致重要的设备失去电力供应,造成严重的后果。因此,对于这些关键设备,应选择可靠的电源方案以确保备用电源的稳定性和可靠性。



实际案例：

某医院在电力中断后，重要医疗设备无法正常运行。经过调查发现，该医院未配备备用电源，导致重要设备失去电力供应。

应对措施：

对于关键设备和重要场合，应选择配备备用电源的方案。备用电源可确保在电源故障或电力中断时，设备能够保持正常运行。在选择备用电源时，应考虑其电力容量、供电时间和可靠性等因素，以确保设备在各种突发情况下能够持续稳定运行。

电源设计余量

在电源设计中留有一定的余量是必要的。如果电源布局设计过于紧凑或负载计算偏低，可能会导致电源过热或过载，从而影响产品的稳定性和可靠性。因此，在电源设计过程中，应充分考虑负载需求、散热条件等因素，合理选择电源的规格，以确保产品在各种工况下的稳定运行。



实际案例：

某服务器在运行高负载任务时出现性能下降的问题。经过调查和分析，发现电源在设计时预留的余量过小，导致电源过载而影响服务器的性能。

应对措施：

在电源设计时留有适当的余量是必要的。通过合理计算负载需求和预留适当的余量，可降低电源过载的可能性，提高产品的稳定性和可靠性。在关键设备和重要场合，应选择具有更大余量的电源设计，以确保设备在高负载情况下仍能保持稳定运行。

其他注意事项

除了选择合适的电源，以下是进一步的应对措施：

1. 电源电路优化

优化电源电路设计可提高电源的效率和稳定性。通过合理设计电路结构、选择合适的元件和采取散热措施等手段，可降低电源内部的功耗和温升，提高电源的可靠性和寿命。

2. 电源检测与监控

为了及时发现电源故障或异常情况，可在电源系统中增加监控装置。通过实时监测电源的电压、电流、温度等参数，可及时发现异常并采取相应的处理措施，避免对设备造成损坏。

3. 定期维护和检查

定期对电源进行维护和检查可及时发现潜在的问题并采取相应的措施。例如，可定期清理电源内部的灰尘和污垢，检查电源线路是否松动或老化，以确保电源的稳定运行。

4. 培训与知识更新

针对电源方面的知识和技能进行培训和知识更新，可提高开发人员和维护人员的专业水平和工作效率。通过了解电源的基本原理、常见问题及解决方案，他们可更好地应对各种电源相关的问题，减少产品降成带来的不良后果。

总结

产品降成从电源上下刀子是一种得不偿失的做法，我们需重视电源的选择和使用。通过合理选择合适的电源、优化电路设计、增加备份或冗余措施、实时检测与监控、定期维护和检查以及培训与知识更新等手段，可有效地提高产品的性能、可靠性和安全性，避免因电源问题而引发的产品故障和损失。



6W隔离宽电压输入电源模块

[点击购买](#)

ZLG嵌入式笔记(连载06) | Linux的RTC回到了1970年，是时光倒流了么？

ZLG 致远电子 2024-12-17 11:34:41

RTC 是一种用于记录时间的电子设备，广泛应用于各种嵌入式系统中。本文通过实际案例分析 RTC 在实际应用中的问题并给出实际建议，如时间“归零”、RTC 时间误差过大等。

从仓库发货的全新产品，为何RTC时间是1970年？

全新产品，开机后系统时间是 1970 年，有两种可能性：

1. 产品 RTC 电池没有正确安装，或者锂电池与电池仓之间的隔离塑料垫片没有去掉，RTC 没电不工作。
2. 安装了 RTC 电池，但 RTC 电池已经被耗尽，这情况一般出现在较长时间库存产品上。

提示：RTC 独立供电的产品，使用前必须正确安装 RTC 电池，并在第一次开机的时候设置正确的 RTC 时间。库存产品，不安装 RTC 电池，或者在安装电池时做好供电隔离。

产品运行一段时间后，RTC变成1970年了，引起程序运行异常

正常运行的系统，经过一段时间后内核出现如下提示：

```
rtc-pcf8563 0-0051: low voltage detected, date/time is not reliable.
```

说明 RTC 电池电压已经偏低。这样会导致 RTC 时间不可靠，从而影响到 Linux 系统时间的准确性。

PCF8563 芯片具有电池电压检测功能，当电压低于 0.9V 时将无法保证时钟信息的准确性（参考图 1），进而导致系统运行异常。

8.4.1.1 Voltage-low detector and clock monitor

The PCF8563 has an on-chip voltage-low detector (see Figure 7). When V_{DD} drops below V_{low} , bit VL in the VL_seconds register is set to indicate that the integrity of the clock information is no longer guaranteed. The VL flag can only be cleared by using the interface.

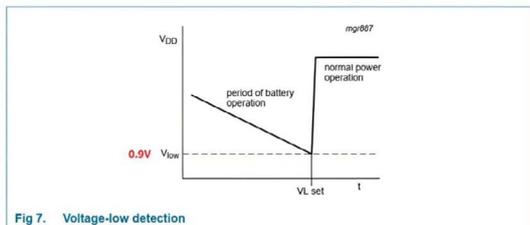


图1 PCF8563 RTC芯片低压检测功能

为了避免出现 RTC 电池电压偏低的情况，可以做如下方面的考虑：

- 可考虑用可充电的 RTC 电池，并设计充电电路；
- 关闭 RTC 芯片的 CLKOUT 功能，降低 RTC 功耗，延长 RTC 电池续航时间；

- 增加超级电容，正常上电时由系统电源供电，系统掉电后由超级电容供电，超级电容耗尽再用锂电池供电。

图 2 是 RTC 多电源供电参考原理图。

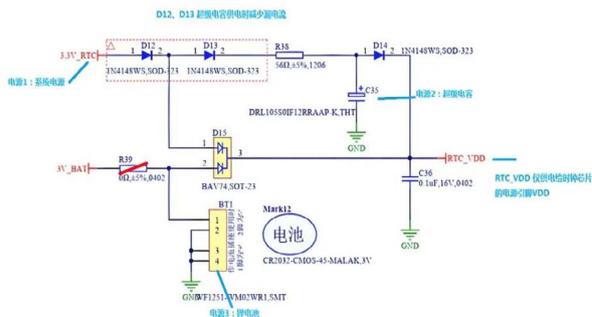


图2 RTC多电源供电参考原理图

设计时需注意以下几点：

- RTC_VDD 仅供电给时钟芯片的 VDD 引脚；
- RTC_VDD 供电线路上使用的二极管选型低耐压、低漏电流；
- I2C 上拉电阻使用系统电源；

预留 CLKOUT 信号测试点，用于调试时钟精度，调试结束后关闭该功能可减少功耗。

让人头疼的“2038年”问题

先介绍一下 UNIX 世界的时间，Unix 纪元时间是从协调世界时 (UTC) 1970 年 1 月 1 日 0 点开始算起，到现在的总秒数。这个时间也被称为 POSIX 时间。Linux 沿用 UNIX 时间，在 32 位处理器的 Linux 系统里，定时器是 32 位的，最大计数是 0xFFFFFFFF，而在 Linux 系统中，rtc_time 定义的是秒、分、小时等都是带符号整数。

```
<uapi/linux/rtc.h>
```

```
struct rtc_time {
    int tm_sec;
    int tm_min;
    int tm_hour;
    int tm_mday;
    int tm_mon;
    int tm_year;
    int tm_wday;
    int tm_yday;
    int tm_isdst;
};
```

32 位定时器有符号整数最大计数为 0x7FFFFFFF，换算成十进制是 2147483647。

2147483647 秒，换算成年的话，是 68.09625973490614 年，大概是 68 年零 18 天。所以，UNIX 时间，从 1970 年 1 月 1 日零时起，经过 68 年零 18 天后（确切是 2038 年 1 月 19 日中午 11:14:07），计数器溢出，RTC 将无法正常工作。在 32 位系统上解决这个问题，需要升级到 Linux 内核到高版本，同时升级 glibc，工作量很大，对一些发布较久的处理器，原厂不一定能提供新版本内核，这是很让人头疼的。如果选用 64 位处理器，跑 64 位 Linux 系统就没这个问题了。

升级到 64 位 Linux 系统后，RTC 计数器最大值为 0x7FFFFFFFFFFFFFFF，换算成十进制是 9223372036854775807。9223372036854775807 秒，换算成年，大约是 292 亿年，完全不用去考虑 RTC 时间溢出的问题了。



ZLG创新性CPM核心板

 点击购买

ZLG嵌入式笔记(连载07) | RTC时钟偶发性延时或超时该怎么办?

ZLG 致远电子 2024-12-18 11:34:02

嵌入式系统运行时，RTC 时钟受多种因素干扰致延时或超时，影响系统时间同步与功能稳定。本文将提出从硬件适配到软件算法优化的综合性方案，以解决此问题，保障 RTC 时钟的精确性与可靠性。

引起延时和超时的主要原因是计时系统使用的基准频率 32.768Khz 精度不够。

MCU 片内自带的 RTC 功能电路由于受到芯片集成工艺的影响一般计量精度都比较差而且功耗也比较大，一般在对时钟精度要求比较高的情况下建议直接使用外部 RTC 专用时钟芯片，比如 PCF8563/PCF85063，下表列出了不同 RTC 芯片的时钟精度。

芯片型号	PCF8563	PCF2129	SD2403
时钟频率(kHz)	32.768	32.768	32.768
时钟精度(ppm)	0.2	3	5

图1 不同RTC芯片的时钟精度

在选择高精度的 RTC 时钟芯片后，还应注意 32.768Khz 晶振的匹配电容选型是否符合晶振设计手册的要求。

[3] Electrical-Characteristics			Measurement temperature :25±2°C	
Item	Symbol	Specifications	Conditions	
1 Nominal frequency	f _{nom}	32.768 kHz		
2 Frequency tolerance	f _{tol}	± 20 × 10 ⁻⁶		
3 Load capacitance	C _L	12.5 pF	晶振回路 总的容值	

图2 晶振的匹配电容

负载电容 CL 的计算公式:

$$CL = \frac{CL1 * CL2}{CL2 + CL2} + Cstray$$

其中: CL1、CL2: 晶振匹配电容; Cstray: 电路板杂散电容

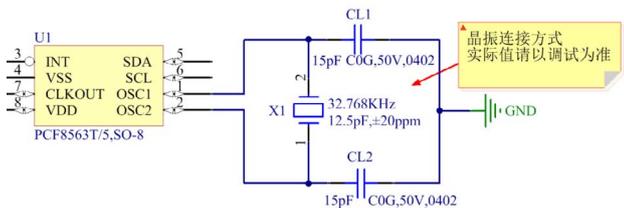


图3 晶体的连接方式及器件参数

代入数值: $CL = [(15 * 15) / (15 + 15)] + 5 = 12.5 pF$

以上是通过硬件电路设计的优化来提高时钟精度，但由于生产工艺、芯片差异等其它不可控因素的存在，实际的时钟精度会存在一定偏差，在时钟频偏稳定的情况下还可以通过软件方式进一步对时钟精度进行补偿，即通过设置 RTC 内部的补偿寄存器来补偿时间。

补偿寄存器的基本原理是通过增加或者减少时钟的修正脉冲来补偿时间而不是改变晶振的频率，如图 4 所示，补偿寄存器的 bit7 设置补偿的模式，bit[6:0] 设置补偿的值。

Table 12. Offset - offset register (address 02h) bit description

Bit	Symbol	Value	Description
7	MODE	0/1	normal mode: offset is made once every two hours
		1	course mode: offset is made every 4 minutes
6 to 0	OFFSET[6:0]	see Table 13	offset value

[1] Default value.

For MODE = 0, each LSB introduces an offset of 4.34 ppm. For MODE = 1, each LSB introduces an offset of 4.069 ppm. The offset value is coded in two's complement giving a range of +63 LSB to -64 LSB.

图4 补偿寄存器描述

1. 补偿值计算方法一

计算流程如图 5 所示。

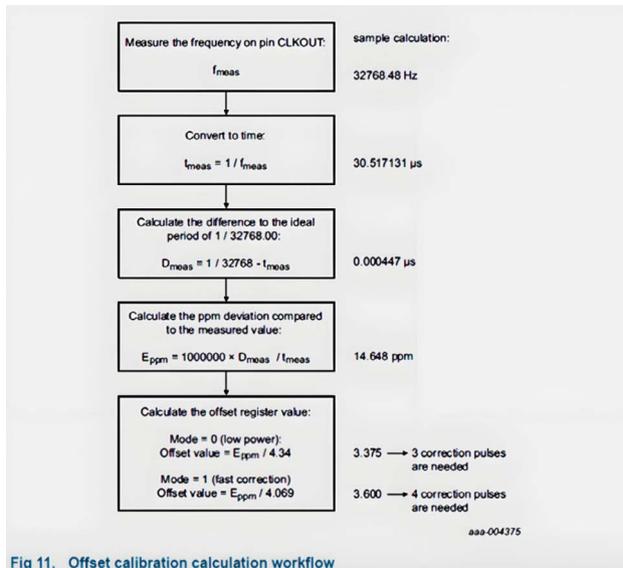


图11. Offset calibration calculation workflow

图5 补偿值计算方法一

- 先用频率计或者高精度的示波器测量 PCF8563 芯片 pin7 脚 CLKOUT 的频率 F_{meas} (软件要先设置 pin7 输出才可测到波形) ;
- 转化成时间 t_{meas} ;
- 计算和理想时钟 32.768KHz 周期的偏差 D_{meas} ;
- 根据测量值计算 ppm 的偏移量 ;

5. 根据设置模式计算补偿寄存器的值，如图中 32768.48Hz 计算得到的 ppm 偏移量是 14.648ppm，如果是模式 0，则算出来的补偿值为 3.375，四舍五入取值 3。

根据补偿计算方式的算式整合简化，若测得实际频率 F 实测值，ppm 的偏移量 Eppm 可以用以下公式快速计算：

$$Eppm = \frac{F \text{ 实测值}}{32768 - 1} * x106$$

如上补偿校准计算框架图 32768.48Hz 算出来的值也是 14.648。

由此可见，若 F 实测值 \geq 32768Hz，则算出的补偿值 \geq 0；若 F 实测值 $<$ 32768Hz，则算出的补偿值 $<$ 0。

2. 补偿值计算方法二

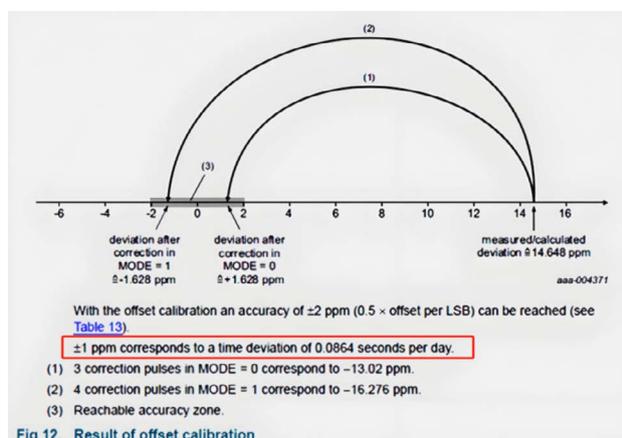


图6 补偿值计算方法二

如图 6 所示， ± 20 ppm 的晶振频率每 1 ppm 的偏差会导致每天偏差 0.0864 s。

在没有测试条件测频率的情况下，可以先测试 24h 下的时间偏差，时间走的快补偿正值，时间走的慢补偿负值，比如某个 RTC 24 小时下测得比北京时间快 7s，那么先算出 ppm 的偏移量 $Eppm = 7/0.0864 = 81.0185$ ppm，然后算出模式 0 下补偿寄存器的值为 $81.0185/4.34 = 18.67$ ，取值 19；算出模式 1 下补偿寄存器的值为 $81.0185/4.069 = 19.9119.91$ ，取值 20。

再比如某个 RTC 24 小时下测得比北京时间慢 7s，那么先算出 ppm 的偏移量 $Eppm = 7/0.0864 = 81.0185$ ppm，然后算出模式 0 下补偿寄存器的值为 $81.0185/4.34 = 18.67$ ，取值 -19；算出模式 1 下补偿寄存器的值为 $81.0185/4.069 = 19.91$ ，取值 -20。

注意：0.0864 s 的计算方式为：ppm 的单位是百万分之一，20ppm 一天的偏差描述为 $(24 \text{ 小时} * 60 \text{ 分} * 60 \text{ 秒} * 20) / 106 = 1.728s$ ，所以 1ppm 一天的偏差时间为 $1.728s/20 = 0.0864$ s。其他精度的晶振也可以用此类方式推算。

如果产品对 RTC 精度要求较高，使用常规 RTC 通过以上校准 / 补偿仍达不到产品要求，就需要考虑选用高精度的 RTC 芯片，如 RX8025T。



ZLG嵌入式笔记(连载08) | 那些不能随便添加的元器件

ZLG 致远电子 2024-12-19 11:32:14

在电路设计中，为了降低成本而忽视 ESD 防护和隔离设计可能导致严重后果。本文将探讨因不当添加元件而引发的电路问题，并逐一解析这些问题，提供实用的解决方案，以确保电路设计的可靠性和稳定性。

电路设计中，有的为了降低成本，没有进行任何 ESD 防护和隔离设计，这是不行的，为了提高可靠性和稳定性，增加保护器件，或者增强驱动能力是常用方法。但稍有不慎，就有可能适得其反，在一些关键电路上，增加一些元件后，会带来很大的负面影响：

1. 如果添加的元件对电路时序产生了影响，这样的器件最好不要加。
2. 在总线时钟信号线上增加了 ESD 保护器件，但器件选型没做好容性控制，这样会引起时钟信号波形畸变，影响正常通信；
3. 在总线源端加 22Ω 电阻也是常用设计手段，但这一定要考虑总线整体电路。如果核心板上已经放置了 22Ω 的电阻，则在底板上则不需要再串联电阻，否则也会引起工作异常。

开机阶段闪屏

1. 现象描述

仪器设备在上电后在 BootLoader 和内核两个阶段出现闪屏现象。先从软件方面着手，通过延迟打开背光，但并未完全解决，通过示波器观察，背光控制部分有明显延迟。再从硬件着手分析，拿到的设备背光控制电路如图 1 所示，通过示波器观察，在设备上电时 LCD_BK 引脚会有一个高电平脉冲，之后维持低电平，最后阶段为 20KHz 的 PWM 波。

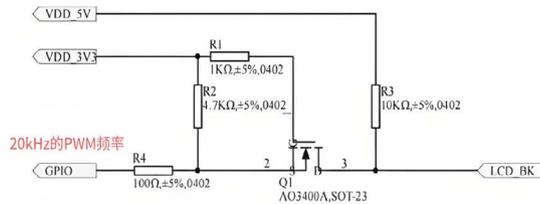


图1 客户的LCD背光控制电路

2. 分析过程

根据以上现象基本可以判断为上电时序问题，通过测量也可以发现 VDD_5V 早于 VDD_3V3 上电。当 VDD_5V 上电时，VDD_3V3 未上电，系统没有启动，GPIO 处于高阻态，因此 Q1 的 1、2 引脚电平相同，Q1 处于关断状态，LCD_BK 信号会被 R3 拉高，导致背光被点亮，而当系统启动后，GPIO 会输出低电平，控制背光关闭，由此出现闪屏现象。

查看背光驱动芯片手册，使能引脚高电平阈值为 1.5V，如图 2 所示，无需进行电平转换，元件 Q1 是多余的，直接使用 3.3V GPIO 既可驱动。

V_{IH}	SHDN Logic High	Enable Threshold Level	0.4	0.8	1.5	V
V_{IL}	SHDN Logic Low	Shutdown Threshold Level		0.7		

图2 使能引脚高电平阈值

3. 解决措施

去掉 R1、R2、R3 和 Q1，并且把 Q1 的 2、3 引脚短接，设备上电启动时不会再出现闪屏现象。

TF卡无法识别

1. 现象描述

这种现象比较常见于 TF 卡所有信号线都加上了 ESD 保护器件的情形，常见问题为 ESD 器件选型不正确。

2. 分析过程

以图 3 所示的原理图为例进行分析。

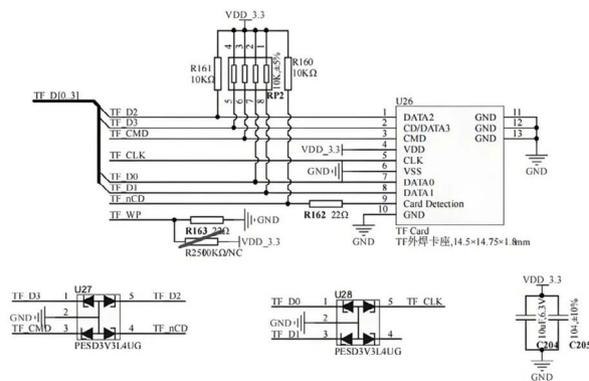


图3 TF卡一般电路图

SD2.0 对 ESD 器件寄生电容的要求为小于 9pF，首先查看该 ESD 器件 (PESD3V3L4UG) 数据手册，发现该 ESD 器件寄生电容最高可到 28pF (参见图 4)，超过了 SD2.0 对寄生电容的要求。

C_i	diode capacitance PESD3V3L4UG	$V_a = 0V, f = 1MHz$	-	22	28	pF
		$V_a = 5V, f = 1MHz$	-	12	17	pF

图4 PESD3V3L4UG的寄生电容

3. 解决措施

断开 CLK 线上的 ESD 器件，或者换用小寄生电容的 DT1446-04S-7，该器件寄生电容仅为 0.65pF (参见图 5)，替换上去后 TF 卡正常识别。

Channel Input Capacitance	$C_{iO} @ V_{IS}$	-	0.55	0.65	pF	$V_a = 2.5V, V_{CC} = 5V, f = 1MHz$
Channel Input Capacitance	$C_{iO} @ V_{SS}$	-	0.65	-	pF	$V_R = 2.5V, V_{CC} = floating, f = 1MHz$

图5 DT1446-04S-7的寄生电容

以太网匹配电阻

1. 现象描述

核心板上以太网 TX_CLK 信号线已有 22Ω 的源端电阻，客户在底板也加了 22Ω 的匹配电阻，引起以太网通信不稳定。

2. 分析过程

将同一块核心板插到评估板上，以太网通信正常，基本定位是底板原因。而后检查底板以太网电路原理图，发现 TX_CLK 引脚串接了 22Ω 匹配电阻。而致远电子官网给出的以太网参考电路里没有加匹配电阻，显然用户没有按照参考电路来设计。

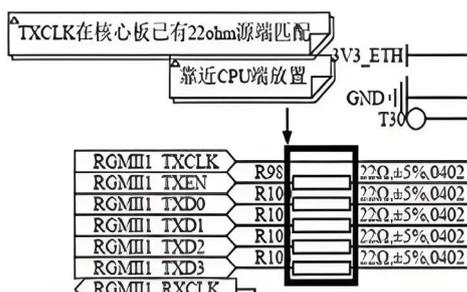


图6 新能源

3. 解决措施

将底板的 22Ω 电阻换为 0Ω，以太网即可正常通信。

4. 总结

以太网 PHY 和处理器端的数据线和控制信号要注意阻抗匹配，避免信号反射。在设计原理图时，一般建议源端串联 22~33Ω 电阻。但是，部分核心板已经在源端串接匹配电阻，设计底板时就不需要串接匹配电阻。使用核心板开发新产品时，建议参考致远电子官方硬件设计参考电路。



ZLG创新性CPM核心板

[点击购买](#)

ZLG嵌入式笔记(连载09) | 电平匹配问题，简单却容

ZLG 致远电子 2024-12-20 11:32:51

在电路设计中，电平匹配是一个基本要求，但常常被忽视，可能导致设备故障和通信异常。本文将揭示如何避免因电平不匹配导致的设备故障，并提供实用的设计建议，确保您的电路设计既高效又稳定。

电平匹配是进行电路设计时的一个基本要求，但有时候却很容易被忽略，特别是没有仔细查看外设或者芯片规格书的时候，根据以往经验来设计，带来了一些问题。这类问题只要在设计的时候稍加检查就能被发现，下面的案例1和案例2都是因为没有仔细看CAN隔离模块的电气特性引起的设计错误，其中案例2问题比较隐蔽，还涉及到不同温度下电气特性的变化：

- 1.MR6450 外接 5V CAN 收发器，工作异常；
2. 使用 CAN 收发器出现偶发性工作异常。

电平匹配在新处理器系统设计时越来越值得注意，现在很多处理器都分多个电平域，有的 IO 是 3.3V，有的是 1.8V。电平不匹配会工作不正常，还会带来损坏 IO 的危险，例如产生电流倒灌而损伤 IO 口。

IIC 电平匹配也是一个经常遇到的问题，下面我们将给出两种不同方案供参考。

TTL和CMOS电平参数定义

我们先看下 TTL、CMOS 逻辑电平参数定义，参考图 1。

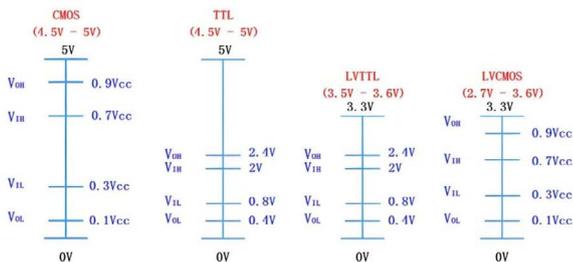


图1 TTL和CMOS逻辑电平定义

不同逻辑电平的两个器件要想实现信号可靠传输的前提条件：

- 驱动器输出的 VOH(MIN) 必须大于或等于接收器输入的 VIH(MIN)。
- 驱动器输出的 VOL(MAX) 必须小于或等于接收器输入的 VIL(MAX)。
- 驱动器输出的电压不得超过接收器输入的 I/O 电压容差。

案例1: MR6450 的UART脚接到了5V的收发器

某客户使用我司 MR6450-L 核心板通过 UART 通讯方式外接 RS485 模块，无法正常工作，与客户沟通后了解到客户使用的 RS485 模块逻辑电平为 5V。

MR6450 核心板处理器为先恒 HPM6450，其 IO 3.3V 逻辑电平定义如图 2 所示。

符号	参数	最小	典型	最大	单位
VDDIO(1.8V)	IO 电源	1.62	1.8	1.98	V
VDDIO(3.3V)	IO 电源	3.0	3.3	3.6	V
(1.8V)VIL	输入低电平	-	-	0.35*VDDIO	V
(1.8V)VIH	输入高电平	0.65*VDDIO	-	-	V
(1.8V)VOL	输出低电平	-	-	0.45	V
(1.8V)VOH	输出高电平	1.35	-	-	V
(3.3V)VIL	输入低电平	-	-	0.25*VDDIO	-
(3.3V)VIH	输入高电平	0.625*VDDIO	-	-	-
(3.3V)VOL	输出低电平	-	-	0.125*VDDIO	-
(3.3V)VOH	输出高电平	0.75*VDDIO	-	-	-
(1.8V)RPU	上拉电阻	35	54	81	kΩ
(1.8V)RPD	下拉电阻	34	55	91	kΩ
(3.3V)RPU	上拉电阻	19	26	44	kΩ
(3.3V)RPD	下拉电阻	21	29	50	kΩ

图2 MR6450 IO 3.3V逻辑电平定义

RS485 模块 5V 逻辑电平定义如图 3 所示。

输入特性						
参数	符号	条件	最小值	典型值	最大值	单位
输入电压	V _{CC}		3.15	5	5.25	
TXD 逻辑电平	高电平	V _{OH}	0.7V _{CC}	≡	V _{CC} +0.5	VDC
	低电平	V _{OL}	0	≡	0.3V _{CC}	
RXD 逻辑电平	高电平	V _{OH}	I _{CCO} =-1.5mA	V _{CC} -0.4	V _{CC} -0.2	≡
	低电平	V _{OL}	I _{CCO} =1.5mA	≡	0.2	0.4
TXD 驱动电流	I _{TXD}		2	≡	≡	mA
RXD 输出电流	I _{RXD}		≡	≡	10	
TXD 上拉电阻	R _{UP}		≡	10	≡	kΩ
串行接口		V _{CC} =3.3V	3.3V 标准 UART 接口			
		V _{CC} =5V	5V 标准 UART 接口			

图3 RS485模块5V逻辑电平定义

由此可知 HPM6450 TX 信号输出的高电平最大值为 VCC-0.1V=3.2V，而 5V RS485 模块 TXD 输入高电平的最小值为 5*0.7=3.5V，很明显不满足驱动器输出的 VOH 必须大于或等于接收器输入的 VIH(MIN) 这一必要条件，IO 口逻辑电平不匹配，从而导致两个模块间无法通讯。

解决方案:

选型 3.3V 逻辑电平的 RS485 模块，或 MCU 和模块间增加电平转换电路。

案例2: CAN隔离模块电平与处理器不匹配，工作时偶尔出现间歇性通讯故障

具体表现为客户使用我司一款隔离 CAN 收发器模块，常温下偶尔出现个别产品间歇性通讯故障，将异常产品置于高温环境下 (65°C) 进行重复上电测试，可复现异常现象。

常温下，测试异常产品的以下信号：MCU 供电、TXD、CAN 差分、CAN 模块供电，正常时各点波形如图 4 所示。

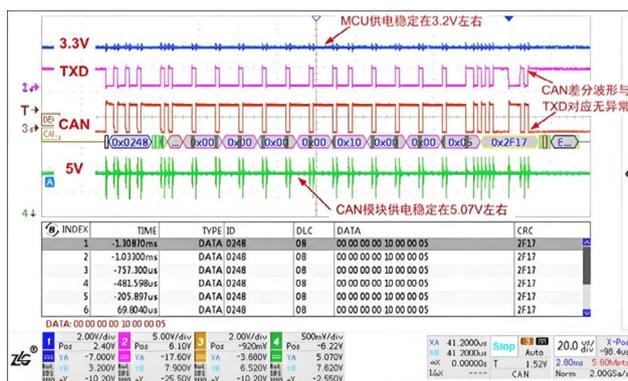


图4 正常时MCU、CAN差分信号、CAN模块供电电平波形

进一步进行高温实验。将异常产品置于65°C的烤箱中，并进行重复上电，测试以下信号：MCU供电、TXD、CAN差分、CAN模块供电，异常时各点波形如图5、图6所示。

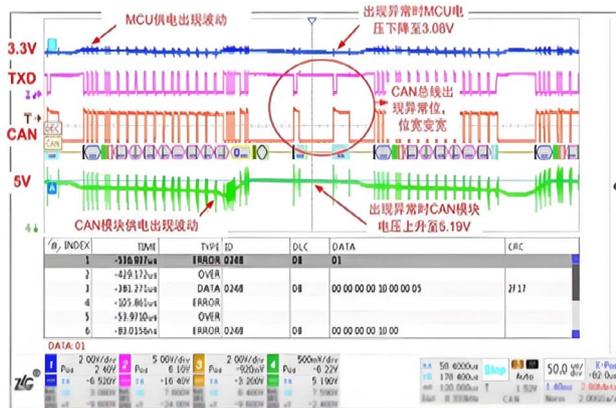


图5 异常时MCU、CAN差分信号、CAN模块供电电平波形

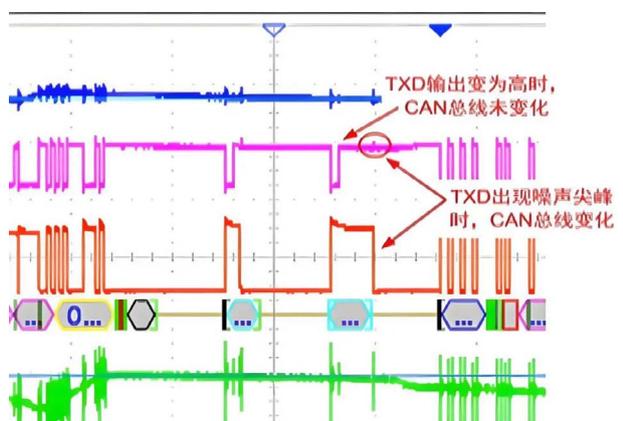


图6 异常时CAN差分信号、CAN模块供电电平波形

分析 CAN 总线出现异常位宽时的波形。当 TXD 信号由低变高时 CAN 总线电平仍然维持显性电平，在偶然间叠加线路板上 noise 后引起 TXD 电平略微变大。此时 CAN 总线电平概率性变为隐性电平。

高温环境下，因半导体材料的特性有可能会使整个系统的功耗有所增加。此案例呈现的现象为 MCU 供电电压下降，CAN 模块供电电压上升，进一步影响到两模块的逻辑电平，从而使异常现象变得更为明显。实测 CAN 模块不同电压和温度环境下，TXD 高电平阈值的变化如图 7 所示。

工作电压	TXD高电平阈值电压 (V)		
	25°C	65°C	85°C
4.75V	2.92	2.88	2.92
5V	3.06	3.02	3.10
5.25V	3.22	3.16	3.22

图7 CAN模块不同电压和温度环境下，TXD高电平阈值的变化

问题根因：

高温 65°C 环境下，CAN 模块供电 5.19V，对应的 TXD 高电平阈值约为 3.17V，MCU 供电 3.08V，对应的 IO 输出电压最高约为 3V，MCU 输出的 VOH 低于 CAN 模块的输入 VIH，两模块间逻辑电平不匹配，导致通讯异常，因 IO 口电平逻辑的容错余量比较小，常温下也许能工作，但在高温环境等比较严格的环境下，两模块间的逻辑电平很容易出现不匹配问题。

解决方案：

将 5V 隔离模块更换成 3.3V 隔离模块。

如果IIC电平不匹配该怎么处理？

1. 使用专用的IIC总线电平转换芯片PCA9306

PCA9306 是一款支持 I2C 总线和 SMBus 的双向电平转换器，支持从 1.0V ~ 3.6V (Vref(1)) 到 1.8V ~ 5.5V (Vbias(ref)(2)) 的电平转换，PCA9306 可工作在两种工作频率下：400KHz 和 100KHz。最大频率取决于 RC 时间常数，一般支持 >2MHz。

PCA9306 标准使用电路如图 8 所示。

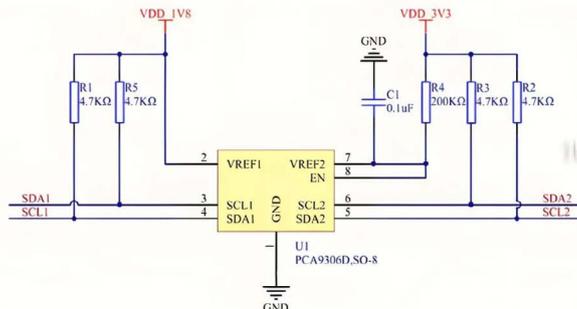


图7 CAN模块不同电压和温度环境下，TXD高电平阈值的变化

设计时注意以下几点：

- 两端的电压 VREF1 和 VREF2 并不是取任意值且 VREF1 为低压位端 VREF2 为高压位端；

边缘计算 ▼

- 可以使用 EN 键来控制内部开关的导通和关断；
- EN 和 VREF2 管脚连接在一起；
- 上拉电阻的取值，取决于 SW 在导通时产生的压降，详见手册推荐值。

2. 使用MOS管搭建电平转换电路

- 图 9 所示电路是一个使用 MOS 管进行电平转换的示例。

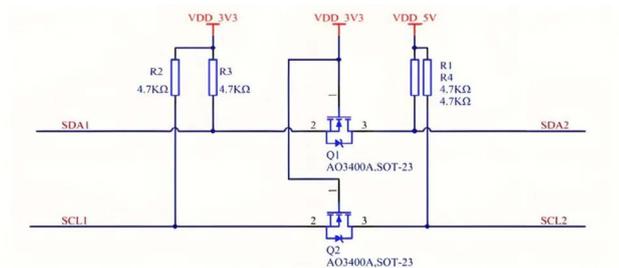
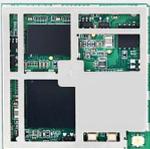


图9 MOS实现电平转换

设计时注意以下几点：

- 低压位 VDD_3V3 连接 MOS 管源极，高压位 VDD_5V 连接 MOS 管漏极；
- 如果换用其它电压阈之间的转换，如 3.3V、2.5V、1.8V 等，需要注意 MOS 管 Vgs 开启导通电压；
- 仅适用于低速率场合（100/400Khz）。



国产芯MR6450

[点击购买](#)

ZLG嵌入式笔记(连载10) | 为什么你的网卡工作会不正常? (上)

ZLG 致远电子 2024-12-23 11:33:07

网卡的正常运作对于系统网络功能的实现至关重要。本文深入剖析因复位时间不足致网卡故障情形，解读不同网卡芯片复位要求及电路设计、驱动配合要点，为工程师提供网卡设计调试参考，助力保障网络系统稳定。

ZLG 致远电子自从设计 ARM 核心板以来，服务了大量客户，其中有很大一部分工作是为客户调试网卡电路和驱动。既有电路设计的问题，也有驱动软件的问题，还有系统网络配置的问题。有时候解决一个问题，并不是单纯硬件或者驱动就能解决，需要多方面配合。就单看硬件方面，除了原理图设计，还有 PCB 布线问题。下面给出几个实际案例，从硬件设计、PCB 布线以及驱动配合等方面进行阐述：

1. PHY 复位时间不够引起网卡工作不正常；
2. MDIO 源端没串 22Ω 终端电阻，信号反射导致网卡工作不正常；
3. PCB 走线没做好等长设计，引起网卡工作异常。

复位时间不够导致网卡工作异常

有一个客户选用致远电子的核心板设计新产品，搭配了自己常用的以太网 PHY 芯片 LAN8720，但是工作不正常，找到我们协助进行调试。经了解，客户原先网卡的复位是 CPU 控制，而在新设计中，参考了我们的开发板，改成了 RC 延时复位。但客户在设计时并未注意网卡芯片的复位时间要求。

1. 详情分析

LAN8720 要求复位时间不得小于 100us（参考图 1），而参考板使用的 DP83848 要求的复位时间是 1us（参考图 2），所以采用的 RC 延时参数很小。在 LAN8720 电路上照搬了 DP83848 的参数，复位时间远不满足器件要求，所以网卡不能正常工作。

Table 6.11 Reset Timing Values

PARAMETER	DESCRIPTION	MIN	TYP	MAX	UNITS	NOTES
T11.1	Reset Pulse Width	100			ns	
T11.2	Configuration input setup to nRST rising	200			ns	
T11.3	Configuration input hold after nRST rising	10			ns	
T11.4	Output Drive after nRST rising	20		800	ns	20 clock cycles for 25 MHz clock or 40 clock cycles for 50MHz clock

图1 LAN8720复位时间要求

DP83848H, DP83848J, DP83848K, DP83848M, DP83848T
SN65250E - MAY 2009 - REVISED APRIL 2015 www.ti.com

4.7 Reset

SIGNAL NAME	TYPE	PIN #	DESCRIPTION
RESET_N	L PU	23	RESET: Active Low Input that initializes or re-initializes the DP83848. Asserting this pin low for at least 1 μs will force a reset process to occur. All internal registers will re-initialize to their default values as specified for each bit in the Register Block section. All strap options are re-initialized as well.

图2 DP83848的复位时间要求

图3是客户所使用的网卡电路。

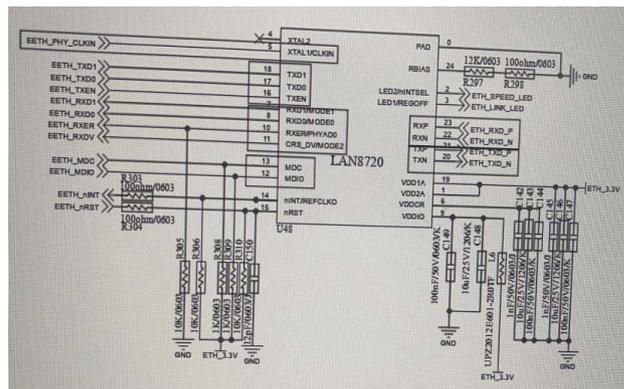


图3 客户使用的LAN8720电路图

从电路图可以看到，RC 复位电路，R=10KΩ，C=12pF，充电时间 $t = RC$: $(10^4 * 12 * 10^{-12}) = 12 * 10^{-8} = 12us$ 。

复位时间 $T = K * R * C$ (K 等于 3-5)，乘以 K 常数，复位时间也不到 100us，无法满足 LAN8720 的复位时间要求。

务必严格按照网卡芯片要求的复位时间来设计复位电路，确保网卡芯片能正常复位。

2. 常见网卡复位时间

不同网卡芯片对复位时间的要求差异非常大，从微秒到毫秒都有，下面汇总一些常见网卡复位时间要求。

- LAN8720: 100us 起，参考图 4。

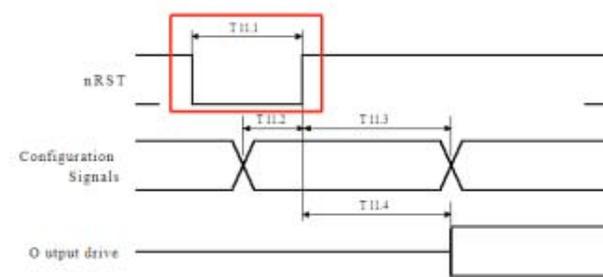


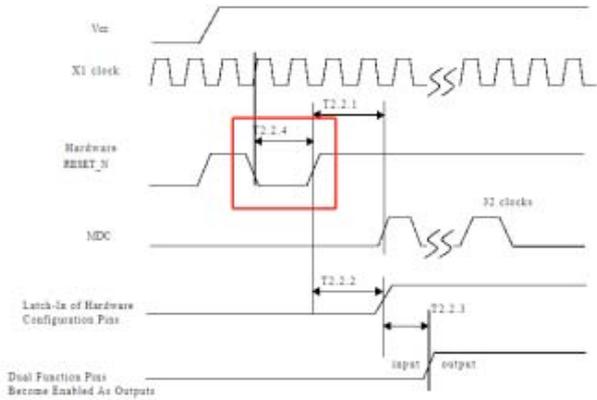
Figure 4.10 Reset Timing Diagram

Table 6.11 Reset Timing Values

PARAMETER	DESCRIPTION	MIN	TYP	MAX	UNITS	NOTES
T11.1	Reset Pulse Width	100			ns	
T11.2	Configuration input setup to nRST rising	200			ns	
T11.3	Configuration input hold after nRST rising	10			ns	
T11.4	Output Drive after nRST rising	20		800	ns	20 clock cycles for 25 MHz clock or 40 clock cycles for 50MHz clock

图4 LAN8720的复位时序和时间参数

- DP83848: 1us 起, 参考图 5。



Parameter	Description	Notes	Min	Typ	Max	Units
T2.2.1	Post RESET Stabilization time prior to MDC preamble for register accesses	MDIO is pulled high for 32-bit serial management initialization		3		μs
T2.2.2	Hardware Configuration Latch-in Time from the Deassertion of RESET (either soft or hard)	Hardware Configuration Pins are described in the Pin Description section		3		μs
T2.2.3	Hardware Configuration pins transition to output drivers			50		ns
T2.2.4	RESET pulse width	X1 Clock must be stable for at min. of 1us during RESET pulse low time.	1			μs

Note: It is important to choose pull-up and/or pull-down resistors for each of the hardware configuration pins that provide fast RC time constants in order to latch-in the proper value prior to the pin transitioning to an output driver.

图5 DP83848复位时序和时间参数

- 裕泰微 TY8512/YT8531, 10ms 起, 参考图 6。

4 OPERATIONAL DESCRIPTION

RESET

YT8512 have a hardware reset pin(RESET_N) which is low active. RESET_N should be active for at least 10ms to make sure all internal logic is reset to a known state. Hardware reset should be applied after power up.

RESET_N is also used as enable for power on strapping. After RESET_N is released, YT8512 latches input value on POS related pins are used as configuration information which provides flexibility in application without mdio access.

YT8512 also provides a software reset control registers which are used to reset all internal logic except some mdio configuration registers. For detailed information about what register will be reset by software reset, please refer to register table.

Table 13. Reset Timing Characteristics

Symbol	Description	Min	Typ	Max	Units
T1	The duration from all powers steady to reset signal release to high	10	-	-	ms
T2	The duration of reset signal remain low timing	10	-	-	ms

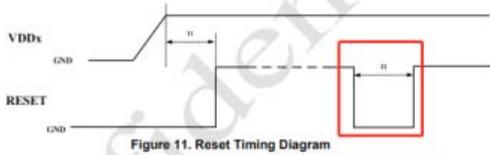


图6 YT8512和YT8531的复位时间要求

- ICPlus: IP101G, 10ms 起, 参考图 7。

Table 14 Reset, Pin Latched-in, Clock and Power Source Timing Requirements

Symbol	Description	Min.	Typ.	Max.	Unit
tclk_lead	X1 input clock valid period before reset released	10	-	-	ms
Tres	Reset period	10	-	-	ms
tclk_MII_rdy	MII RMI clock output ready after reset released (Pins settings are latched into IP101G this moment)	-	-	10	ms
tdiff	Time difference between AVDD33 and VDDIO, DVDD_REGIN	-	-	30	ms
tpwr_lead	All power source ready before reset released	11	-	-	ms
tlat_s	Latched-in pin signal setup time	10	-	-	μs
tlat_h	Latched-in pin signal hold time	2.63	-	-	ms

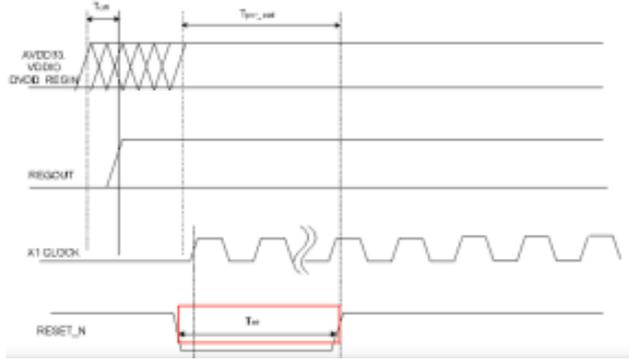


图7 IP101G的复位时间和时序

综上所述, 在一般设计中, 建议采用主控芯片的 GPIO 来控制 PHY 芯片的复位, 并在靠近 PHY 芯片的附近并联 RC 电路, 增强稳定性; 对于主控芯片 IO 资源紧张, 可以采用 RC 延时复位, 但务必计算好需要的复位时间, 并预留充足的裕量。

另外满足了这个手册要求的复位时间也不一定就能够正常工作。设计中如果采用 MAC 给 PHY 提供参考时钟的工作方式, 由于主控的参考时钟默认可能是输入模式, 需要加载驱动后将软件配置为输出模式, 所以还要保证复位释放前, 以太网驱动已经加载完毕, 设计 RC 复位参数时需要将这些时间都计算在内。

其实不仅仅是网卡芯片, 其它带复位引脚的芯片, 在设计电路的时候都需要考虑芯片的复位时间要求, 以设计满足条件的电路, 确保其能正常工作。



M62xx核心板

点击购买

ZLG嵌入式笔记(连载11) | 为什么你的网卡工作会不正常? (中)

ZLG 致远电子 2024-12-24 11:32:32

网络故障困扰重重? 本文聚焦客户网卡故障的两种特殊情形: 低温运行与信号干扰下的网络卡死。看似简单却历经波折的解决过程, 带你深入剖析排查中陷入的误区。

MDIO信号反射导致网络卡死分析

客户反馈网卡故障表现有两种情形:

1. 在低温情况下运行一段时间后, 容易出现网络卡死, 必须重启才能恢复;
2. 外界稍微有点信号干扰就有可能引起网络卡死。

问题的解决方法看似很简单, 但解决过程却颇费周章, 花费了很多的精力。排查出原因是 MDIO 的时钟信号在低温情况振铃增强或受到干扰时, 信号产生毛刺, 导致寄存器被写入了错误的值, PHY 进入了异常状态无法工作。最终解决的方法是在 MDC 信号靠近源端串联 22 欧电阻。下面进行分析。

首先问题的现象是低温工作一段时间后出现, 双网口的板卡只有其中一个网口会出现该问题, 且网络一旦卡死, 必须重启才能恢复, 所以开始认为是走线问题或电源问题诱发的 MAC 或 PHY 卡死。

另一个陷阱是 PHY 寄存器有两部分, 由于 MDIO 协议定义的 register address 位宽为 5bit, 对 PHY 来说是不够用的, 所以原厂将其分成了一般寄存器和扩展寄存器两部分。

以裕泰微的 YT8531 为例, 通过一般寄存器的 0x1E 和 0x1F 间接地读写扩展寄存器, 地址大于 0x1F 的寄存器都需要通过 0x1E 和 0x1F 作为入口, 参考图 1。初次检查寄存器的时候只读出了一般寄存器的值, 检查对比异常时和正常时没有区别就认为寄存器没有问题了。后期又仔细核对发现出问题时在扩展寄存器, 才又将调试方向纠正过来。测试 MDIO 波形发现原本应该在初始化过程中配置好的地址, 正常通信过程中也在不停的覆写, 最终在反复的“尝试”后, 终于等到 MDC 毛刺, 成功将错误值写入了寄存器。最终也通过示波器测到了 MDC 上的毛刺, 测量得到的波形如图 2 所示。



图1 示波器协议解析功能测量MDIO波形

MDIO 是以太网电路中十分重要的部分, 配置模式、调整性能都离不开 MDIO, 很多以太网故障都与 MDIO 有关。MDIO 协议规定是可以多个网口使用同一个总线的, PHY 芯片基本也都留有设置 PHY 地址的上下拉电阻, 但很多主控芯片原厂还是会不建议一对多的设计。MDIO 看似和 I2C 很像, 但实际还是与 I2C 有很大区别, 它的协议时序不同, 速率也比 I2C 高很多, 没有应答位, 所以 MDIO 的上拉电阻不能用阻值太小的, 信号走线也需要注意抗干扰和阻抗匹配。

关注我们, 下期继续讲解: 以太网 Link Down 问题排查题排查及其他容易忽略的问题。



M62xx核心板

[点击购买](#)

Motorcomm YT8531SH / YT8531SC Datasheet



6.2.20. Rx Error Counter Register (0x15)

Table 53. Rx Error Counter Register (0x15)

Bit	Symbol	Access	Default	Description
15:0	Rx_err_counter	RO SWC	0x0	This counter increase by 1 at the 1st rising of RX_ER when RX_DV is 1. The counter will hold at maximum 16'hFFFF and not roll over.

6.2.21. Extended Register's Address Offset Register (0x1E)

Table 54. Extended Register's Address Offset Register (0x1E)

Bit	Symbol	Access	Default	Description
15:8	Reserved	RO	0x0	Reserved
7:0	Extended Register Address Offset	RW	0x0	It's the address offset of the extended register that will be Write or Read.

6.2.22. Extended Register's Data Register (0x1F)

Table 55. Extended Register's Data Register (0x1F)

Bit	Symbol	Access	Default	Description
15:0	Extended Register Data	RW	0x0	It's the data to be written to the extended register indicated by the address offset in register 0x1E, or the data read out from that extended register.

图1 YT8531扩展寄存器入口

ZLG嵌入式笔记(连载12) | 为什么你的网卡工作会不正常? (下)

ZLG 致远电子 2024-12-25 11:35:17

设备与电脑通信时，自适应速率千兆满载测试现异常。经多方面排查，因 TXD3 多种线路问题叠加致信号完整性受扰，通信失败，还介绍以太网常见故障及其他设计调试重点，经验方法值得借鉴。

以太网Link Down问题排查

这个问题的故障现象是：设备在与电脑通信，开启自适应，速率千兆时，设备端使用 iperf 进行 1000Mbit/s 满载测试，偶发的调试串口会出现 Link down 打印，通信数据大量丢包，然后会马上 link up，有时 Link up 成千兆，通信会恢复正常；有时 Link up 成百兆或者十兆，通信会中断，ping 包不通。手动输入 ifconfig eth0 down, ifconfig eth0 up 后恢复正常 ping 包。

排查过程：

1. 电脑往设备发满载测试，并不会出现 Link down 的问题。
 2. 设备往电脑发，当只是发 100Mbit，即 10% 的负载时，也不会出现 Link Down 的问题。200Mbit 时小概率复现。
 3. 强制成 100M 全双工时，双向发送满载测试，不会出现 Link Down 的问题。
 4. 测量 25M 晶振实际频率为 24.99984M，尝试过飞线更换成 25M 有源晶振，问题还在。
 5. 测量 PHY 芯片输出给 MAC 的 125M 时钟测量为 125M，波形正常。
 6. PHY 芯片的 0V/2.5V/3.3V 供电纹波正常，均小于 20mV。
 7. RGMII_TXD 的 4 根数据线，分别测试使用 0Ω、22Ω 电阻串联，增加 7K 上拉，无改善。
 8. 飞线带变压器的 RJ45 测试，或使用长度较短的 CAT6 网线，均无改善。
 9. 把 PHY 芯片配置的 RGMII 发送和接收时钟的 Clock Skew 都设置为 0，通过 MAC 去调整，测试满载也是会 down。
 10. 使用示波器探头去点 RGMII TX 的数据线时，PING 包会断。放开探针就可以恢复。
 11. 飞线把 RGMII 的 IO 电平改为 1.8V，可以解决这个问题。
 12. 不使用底板的 3V，使用数字电源来供电 3.3V 给底板，问题还在。
 13. 把 TXCLK、TXD0~3、TX_EN 这 6 根线，从连接器出来断开，到 PHY 芯片也断开，直接飞线，不使用底板的走线，验证是可以解决这个问题。
 14. 把串联的电阻改为焊锡直连，问题还在。
 15. 只断开 TXD3 这根，再飞线，可以解决这个问题。
- TXD3 这根线的左右两边分别是 MDC 和 PWM 信号，跟 TXD3 的中间距离最小为 12mil，线粗为 5mil（参见图 1），不满足 3W 规则。尝试割断左右这两根线并焊接飞线，也可以解决这问题。

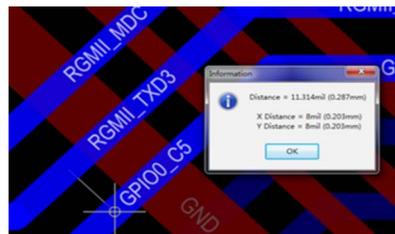


图1 RGMII_TXD3 PCB走线局部

16. 检查 PCB 走线，RGMII 和 LCD 是复用管脚，在 PCB 走线的时候，会有一点分叉存在（最长分叉为 2mm 左右），参见图 2。

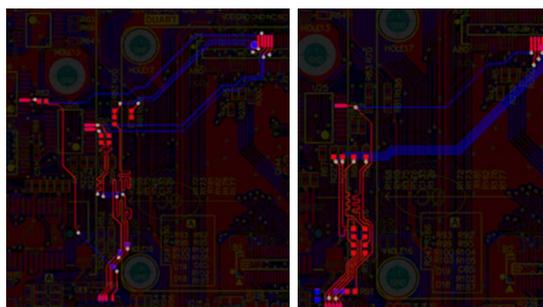


图2 RGMII 底板PCB走线 TX和RX

17. RGMII TX 组走线长度：3010mil~3224mil，RGMII RX 组走线长度：2403mil~2571mil（参见图 3），单端阻抗整板为 50 欧。从等长来看，TXD3 跟其它发送的数据线并没有严格等长，根据手册要求，TXD0~TXD3 是需要按 60mil 以内的等长的，且总长度应在 3000mil 以内。

信号	核心板	底板	总和
RXCLK	1263.038	2401.769	3664.807
RXD0	1263.7	2419.095	3682.795
RXD1	1263.889	2407.04	3670.929
RXD2	1363.88	2412.079	3775.959
RXD3	1308.604	2403.619	3712.223
RXDV	1262.929	2571.564	3834.493
TXCLK	1304.893	3017.724	4322.617
TX_D	1308.965	3020.66	4329.625
TX_D	1321.475	3010.578	4332.053
TX_D2	1376.411	3030.561	4406.972
TX_D3	1419.634	3224.664	4644.298
TX_EN	1314.614	3013.768	4328.382

图3 RGMII PCB 等长数据

经过以上的一系列调试分析可推知，由于 TXD3 没有做好等长，终端电阻没有摆放在靠近源端，TXD3 旁边有时钟线干扰，走线有分叉且总长度太长，多种因素叠加在一起，导致以太网测试满载时会 down。

此问题属于比较典型的信号完整性问题，数字电路中，电信号被规定为 0 和 1 两种电平，使得数字信号有极强的抗干扰能力，所以信号完整性问题往往不会很明显的暴露出来，也许单独有串扰或者有分叉都不会导致最终的通信失败，但多个问题叠加起来就导致电路的时序裕量被耗尽，进而通信失败。

其他容易忽视的问题

复位时间不够导致网卡工作异常、MDIO 信号反射导致网络卡死以及以太网 Link Down 问题都是典型的以太网常见故障，其中调试过程的经验方法值得借鉴，还有一些其他需要重视和警惕的问题，设计和调试时应当做排查重点：

1. 以太网 PHY 到变压器这段的驱动分为电流型和电压型，设计时需要查看手册，确定变压器这边的中间抽头是要拉高还是接地。

2. PHY 的设备地址 PHYADDR 通过上下拉电阻来实现。PHYADDR 往往和 LED 管脚复用，设计时要注意如果 PHYADDR 是上拉，则该管脚为低电平有效，应该接到 LED 的负极；如果 PHYADDR 是下拉，则该管脚为高电平有效，应该接到 LED 的正极。

3. RMII 不像 RGMII 和 MII 有一个 TXCLK 一个 RXCLK，取而代之的是一个 50MHz 的 REFCLK，这个时钟设计是要注意核对主控端和 PHY 端各自是作为输入还是输出，是否可配置，有些厂家的是可以通过软件配置的，有些却是固定的，需要在设计前确定 REFCLK 的方向。

最佳的以太网电磁兼容拓扑应为：PHY 芯片共模电感 / 磁珠 TVS 管网络变压器气体放电管 / 放电齿 RJ45。容易被忽视的是以太网 POE，当连接了支持 POE 的交换机，以太网的信号线上会有最高 57V 的电压，即使产品不需要 POE 功能，也要注意器件的耐压值。



M62xx核心板

[点击图片](#)

ZLG嵌入式笔记(连载13) | 以太网带宽总是跑不满，是什么原因？

ZLG 致远电子 2024-12-26 11:32:16

随着网络需求增长，千兆以太网速率问题备受关注。本文针对其实际速率低的情况，详细分析布线不佳等原因，并从等长走线、阻抗匹配等硬件方面及软件调试角度，提供全面提升速率与稳定性的有效方法。

正常的千兆以太网理论速率可以达到 1000Mbps，实际考虑到通信各个环节的损耗，也应在 800Mbps 以上。而实际上却有很多板卡，按照参考电路设计的千兆以太网实测只只能跑到 300/400Mbps，甚至更低。这种情况 90% 是由于以太网布线不佳导致的信号完整性问题。

千兆以太网 RGMII 信号的速率高达 125MHz，而且是双边沿触发的，所以对信号完整性是有要求的。以下的 RGMII 信号完整性原则和方法，可以帮助优化和改善以太网的速率和稳定性。

1. 等长走线

RGMII 在走线时必须要进行等长处理。等长即 RGMII 总线中的 TX_CLK、TXD[0:3]、TX_EN 这 6 个信号为一组，RX_CLK、RXD[0:3]、RX_EN 这 6 个信号为一组，分别绕蛇形走线，使得每组组内信号线之间长度差值在 100mil 以内，不同厂家的公差要求可能不同，但对硬件工程师来说，条件允许的情况下，公差越小，则时序裕量越大，设计时应尽量将公差缩小。注意这个公差应该是 CPU 的 MAC 到 PHY 芯片的走线，有些核心板的手册标注了核心板 RGMII 走线长度和 CPU 内部的长度，此时必须要把它们和底板的长度加起来计算等长。

2. 阻抗匹配

RGMII 的所有信号线要求单端 50Ω 的阻抗匹配，50Ω 是一个约定俗成，沿用至今的值，也是板厂熟悉，易于叠层和制板的取值。阻抗匹配指的特征阻抗，定义是：传输线长度趋近于无穷大时，传输线的阻抗是多少（例如上述的 50Ω）。在有限长传输线中，它表征电信号经过导体瞬间的阻抗，但这个值不是 50Ω，阻抗匹配即有限长传输线中每一处的特征阻抗都是相等的、均匀的、连续的。设计时，它是由 PCB 的铜厚、线宽、相邻层间距、板材等计算得来的。

对设计师来说，最重要的是阻抗的连续性，电信号经过阻抗不连续的导体，如同光线经过密度不同的介质（水和空气）时产生反射与折射，信号就会失真。传输线打孔、换层、经过电阻和连接器的焊盘、有分叉，都会导致不同程度的阻抗不连续。所以 layout 时应避免分叉、走线改变方向用 45 度折弯，尽量避免打孔换层。

3. 完整的参考平面

参考平面对阻抗匹配是非常重要的，传输线原理中，传输线包含信号线和回流路径，当信号线是均匀连续的，但参考平面不连续也会导致传输线不连续。参考平面是根据叠层确定的，一般是相邻的平面层，例如图 1 中，8 层板叠层 1 层和 3 层参考 2 层，4 层参考 5 层，6 层和 8 层参考 7 层。信号线投影区域内参考层的铜皮也不能中断或改变网络。容易出问题的地方是换层的时候如果信号从 1 层打孔到了 8 层，那么参考层就从 2 层变成了 7 层，建议在信号换层位置 200mil 范围内打地过孔创造回流路径。有条件的地方也应补上地过孔增强两层地的连续性。那么当信号从 1 层换层到 4 层，参考

层由 GND 变成了 PWR，网络都改变了，这时该怎么办呢？这种情况可以在换层位置附近布置 0.1uF 的缝合电容将 2 层和 5 层的网络连在一起。



图1 一种常见8层板叠层结构图

4. 避免源端反射

前面讲过阻抗匹配，当信号传输到接收端，RGMII 的接收 IO 是高阻的，信号就会反射回去，反射波到达源端又发生二次反射，叫做源端反射。源端反射会和原始信号叠加在一起形成过冲、振铃等，影响信号质量。通过在靠近信号发送端的位置串联源端匹配电阻，可以有效降低源端反射。有些芯片推荐的匹配电阻是 33Ω，有些是 22Ω，这是因为芯片管脚驱动器的输出阻抗不同，当驱动器输出阻抗 + 源端匹配电阻的阻值为 50Ω 左右的时候，配合 50Ω 的阻抗匹配，防止源端反射的效果才最好，而这也是串阻要靠近信号源端的原因。

5. 防止串扰

外部的干扰也是不能忽视的，检查诸如走线之间有没有保持 3W 间距，时钟走线有没有包地。类似图 1.44 的叠层中 3 层和 4 层的走线有没有和相邻层的其它走线有长距离、近距离的平行走线等等，在这些地方往往也能找到能够优化的点。

6. 软件 Delay line 调试

如果板子调试时发现速率不够，其实也不一定就是硬件问题，可以先看下软件有没有进行 Delay line 的配置。

Delay line 就是在 CLK 上增加一段无用的 buffer 来达到延时的效果，通过调整 buffer 数量，来补偿 PCB 走线的延时，最终使得接收端接收到的时钟边沿移动到数据眼图的正中间。一般 MAC 端和 PHY 端都可以进行 Delay line 的调节，但 MAC 端调节的精度更细，操作也更方便，一般将 PHY 端的 Delay line 设为 0，只在 MAC 端调节。

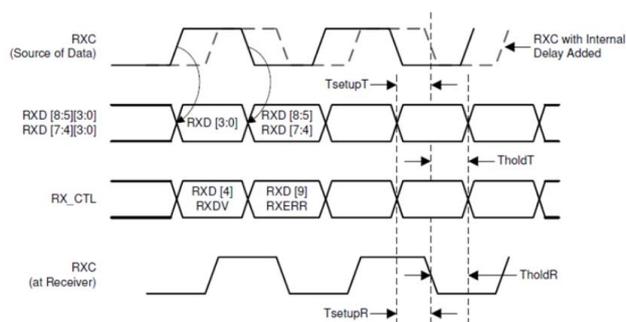


图2 Delay line电路结构示意图

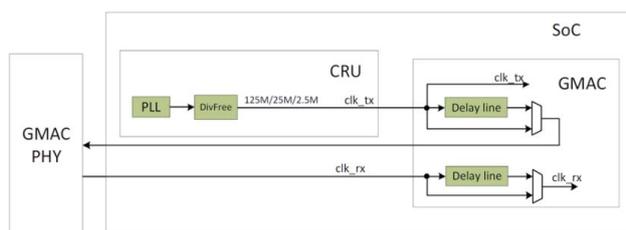


图3 Delay line调整波形示意图

刚开始调 Delay line 时，网络有可能完全 ping 不通，同时没法区分是发送有问题还是接收有问题，所以要借助 PC 端的 Wireshark 先调试发送数据。

MAC 端调节 Delay line 每个步进增加 0.08ns 延时，待调试的板子在后台 ping PC 的 IP，输入命令让板子遍历所有 TX Delay line，此时观察电脑端的 Wireshark，会看到从某个值开始收到板子发来的 ICMP 包，称为窗口的开始；随着 TX delayline 逐渐增加，收到的数据包会变得更完整，接着变得不完整，最后彻底收不到板子发来的 ICMP 包，此时的 TX Delay line 值称为窗口的结束。取窗口开始和结束的中间值作为此 PCB 板的 TX Delay line 最优值，记下这个值。

调通板子网卡的发送功能之后，板子就具备了应答 ping ICMP 包的能力。下面在电脑端绑定板子网卡的 MAC 地址和 IP 地址，并开始 ping 板子的 IP 地址，板子执行以下命令遍历所有 RX Delay line，观察电脑端 ping 命令的返回信息，会看到 ping 返回失败，渐渐变成 ping 成功，而后又变为 ping 失败，记录下这些节点对应的 RX Delay line 值。ping 成功的这段区间称为接收窗口，同样取其中值，作为 RX Delay line 的最优值。最后把 Delay line 最优值设置，添加到设备树的 gmac 节点中，固化 Delay line，完成 Delay line 的配置。

要注意的是，使用 RGMII 接口连接千兆 PHY 时，只要硬件存在差别，都需要重新做一次 Delay line 的配置。

另外可以调整 Delay line 也不代表硬件不需要做等长 Layout；如果不做等长，意味着时钟信号的跳变边沿要同时对四条数据线的相位进行延时微调，它们重叠的区域才能对数据正确采样，但这个区域将会变得过窄，甚至超出 Delay line 延时的范围。

7. 调整驱动强度

除了配置 Delay line，还可以调整 RGMII 的驱动强度进一步改善问题。当使用 iperf 长时间测试发送会掉线也可以尝试调整驱动强度。

当将驱动强度调高时，信号的边沿会变得更陡峭，但与此同时，过冲和振铃也会变强，驱动强度调低时，信号边沿会变缓，但能够减少过冲和振铃的影响。驱动强度通常只有几个档位（通常 4 或 8 档），有些厂商会标出是多少 mA，有些则只标出强度等级，没有具体单位，所以不用像 Delay line 那样找最佳值，实测在不同强度下速率最高，稳定性最好的取值即可。



ZLG嵌入式笔记(连载14) | IIC总线上拉电阻阻值如何确定?

ZLG 致远电子 2024-12-27 11:31:31

I2C 总线在产品设计中被广泛应用，尽管其结构简单，但经常发生上拉电阻设计不合理的问题。本文将对 I2C 上拉电阻的选择进行简要分析。

一根信号线上，通过电阻连接一个固定的高电平 VCC，信号线初始、空闲时一直保持高电平状态，称为上拉电阻。

总线的上拉电阻各有不同作用，如 I2C 的上拉电阻是由 I2C 端口内部结构决定的，而 RS485 和 CAN 总线的上拉电阻是为了做终端匹配。那这不同总线的上拉电阻，是否接上拉，阻值的选择有何讲究吗？

是的，这些电阻阻值的选择都是有理论依据的，如果阻值选择不正确，会引起一些异常，例如曾经有一个案例，I2C 上拉电阻阻值为 10K，而应用却要求 I2C 跑 400KHz 的高速率，显然这样的上拉电阻不能满足 400kHz 的高速率要求，出现通信错误也是不可避免的了。

对于 RS485 和 CAN 总线而言，则需要根据现场情况进行灵活使用，就算节点数相同的两个应用，由于布线和所接设备的差异，需要的匹配电阻也是不同的。理论计算值可做为重要的参考依据。

IIC上拉电阻有什么作用、阻值如何确定?

目的：保证通讯正常。

常说 IIC 需要加上拉电阻，却不知道为什么要加上拉电阻？

第一是 I2C 本身的协议决定，空闲时两线均保持高电平状态，能够实现线“与”功能。

第二是器件内部结构，如图 1 所示，SCL 和 SDA 引脚内部结构，内部无法提供高电平，只能依靠外部高电平。目前部分器件支持上拉功能的复用功能开漏模式，芯片内部可配置为上拉模式（建议使用外部上拉电阻）。

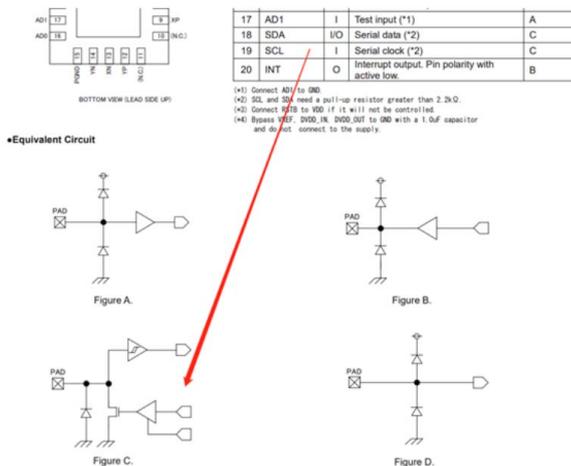


图1 IIC端口内部结构等效电路

IIC上拉电阻的取值大小怎么确定?

考虑 IO 灌电流，阻值不能过小。通常在标准模式 100KHz，灌电流不超过 3mA，

VOL 不超过 0.4V，若上拉电阻至 3.3V，上拉电阻阻值约大于 1k。

$$R_{min} = \frac{V_{dd} - Vol(\max)}{I_{ol}(\max)}$$

代入 Vdd = 3.3, Vol(max) = 0.4, Iol(max) = 3, 得到 Rmin = 966Ω。

考虑上升时间，阻值不能过大。高电平是靠外部上拉电阻提供，电阻和总线电容组成 RC 网络，电阻过大充电时间过长，造成信号上升缓慢，未在规定时间内到达高电平，导致数据错误。针对 I2C 系统，在上拉电压的 30% 至 70% 时来测量上升时间和下降时间，总线电容 100pF，总线最大上升时间 300ns。

$$R_{max} = \frac{Tr}{Cb * x \ln(\frac{7}{3})}$$

代入 Tr = 300, Cb = 100, 计算得到 Rmax = 3.54kΩ。

根据以上计算，上拉电阻的取值在 966~3.54k 之间。工程实际设计中，并不是完全按照这个值来设计，通常可选取 2.2k, 3k, 4.7k。

解决 I2C 通信问题，建议抓取 I2C 波形，观察总线上升 / 下降时间，并分析总线电容和上拉电阻等因素，并结合驱动代码全面分析。



M62xx核心板

点击购买

ZLG嵌入式笔记(连载15) | RS485偏置电阻和终端电阻有什么作用、怎么选取?

ZLG 致远电子 2024-12-31 11:34:21

在工业通信领域，RS485 总线的稳定运行至关重要。本文聚焦其关键的电阻设计问题，以 RSM485PHT 为例，深入剖析偏置与终端电阻的作用及影响，同时提供实用的调整策略和应用指导。

RS485 总线的偏置电阻主要是给 A、B 确定的逻辑状态。

RS485 总线的终端电阻主要是用于信号线的阻抗匹配、提供通信线缆寄生电容能量的泄放路径、提高信号质量。

此次以自收发产品 RSM485PHT 为例进行说明，该产品内置完整的 DC-DC 和信号隔离电路，具有较强的抗扰性和高可靠性，具备自动收发功能。该产品的 A、B 线内置 47kΩ 的上、下拉电阻，收发器输入阻抗最小值为 48kΩ。

本次测试的硬件框图如图 1 所示，在通信速率为 500kbps、总线上挂 6 个通信节点、双绞线总长度约为 3 米。

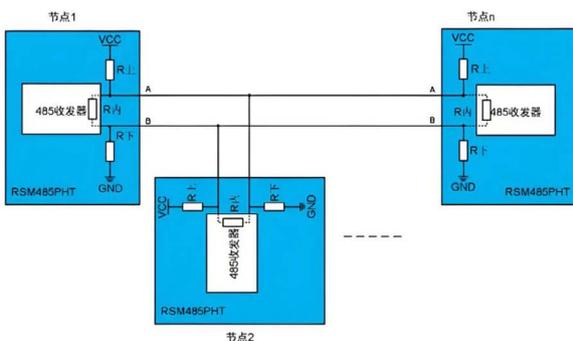


图1 RSM485PHT组网通信框图

抓取总线波形如图 2 所示，逻辑 1 的差分电压约为 1.60V。

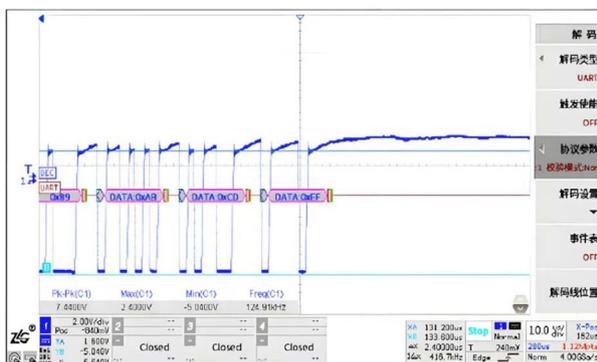


图2 3m双绞线、500kbps时VAB波形

图 3 为本次 RSM485PHT 测试的电阻分压等效图。

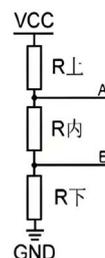


图3 RSM485PHT电阻分压等效图

当总线上有 6 个节点进行通信时，相当于 6 个 R 上、6 个 R 下、6 个 R 内并联，此时 VAB 高电平电压计算值为：

$$VAB = \frac{\frac{R_{内}}{6}}{\frac{R_{上}}{6} + \frac{R_{内}}{6} + \frac{R_{下}}{6}} * VCC$$

取 VCC=5.1V，得到 VAB=1.72V，考虑到线阻的分压，此计算值 1.72V 与实测波形幅值 1.60V 基本吻合。

因为总线端逻辑 1 电平幅值仅有 1.6V 左右，该幅值的抗扰能力相对较弱且影响了通信距离的进一步拉长，现考虑通过外加偏置电阻将总线幅值抬高至 3.5V 左右。通过上面 VAB 计算公式，可计算出 R 上等效 = R 下等效 ≈ 2.75kΩ，外加上下拉电阻值约为 4.1kΩ。如图 4 所示为外接 3.5kΩ 上下拉电阻时总线的通信波形图（外加偏置电阻增加功耗约为 5.1V/3.5k ≈ 1.4mA，在可接受的范围内），因为实际偏置电阻值 3.5kΩ 小于 4.1kΩ，实际总线逻辑 1 幅值为 3.92V 高于预设值 3.5V。

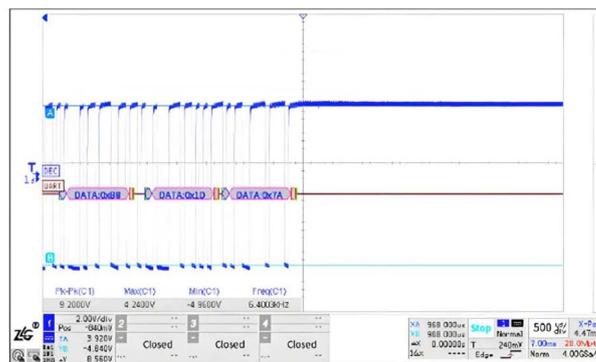


图4 外加3.5kΩ偏置电阻时差分波形图

在上述接入 3.5kΩ 上下拉电阻的环境下再接入 120Ω 的终端电阻，此时电阻分压等效图 3 中的 R 内等效 ≈ 60Ω，将各数值代入 VAB 公式，计算出电压约为 60mV，测试波形如图 5 所示。此时高电平处于门限 -200mV ~ +200mV 门限内，收发器无法识别逻辑 1，造成通信错误。

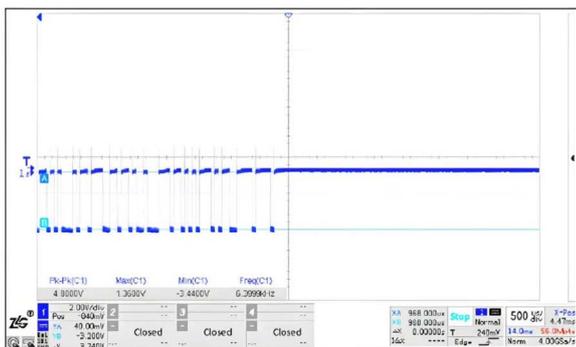


图5 接入120Ω终端时差分波形图

在使用我司自动收发模块 RSM485PHT 或 RSM485M 时，如果总线逻辑 1 电平较低可通过外接偏置电阻来调节总线电平，偏置电阻值过小将增加额外的功耗，电阻值太大调节效果将不明显。

偏置电阻值可以根据实际节点数量算出等效电阻值，再代入阻抗分压公式：

$$VAB = \frac{R_{内等效}}{R_{上等效} + R_{内等效} + R_{下等效}} * VCC$$

其中 VCC 可取 5.1V，VAB 一般取 2.5V ~ 4.0V。

具备自动收发功能的模块 RSM485PHT 或 RSM485M 的总线逻辑 1 电平是通过 AB 线的偏置电阻来驱动，其能力弱于推挽驱动，因此一般情况下我们不建议用户加终端电阻。如果通信速率高、通信距离长，总线信号质量很差，需要加终端电阻来减弱反射信号或提供泄放寄生电容能量的路径，可以选择稍大阻值的电阻，并且可以考虑通过在 AB 线上加小阻值的偏置电阻两者配合来调节总线的电平。

总的来说在使用自动收发 RS485 进行通信时一定要确保 A/B 线差分电压不会处于 -200mV ~ +200mV 范围内；若逻辑 1 或逻辑 0 的差分电平幅值较低，可以通过外加小的偏置电阻来进行调整；一般情况下不建议用户接终端电阻，如果要接尽量选择较大的阻值同时与外接偏置电阻搭配使用。



隔离RS-485收发器 RSM485PHT

👉 点击购买

【产品应用】 ZMC600E运动控制器T型、S型速度规划详解

ZLG 致远电子 2024-12-30 11:37:22

想深入了解 ZMC600E 运动控制器的速度规划吗？T 型线性加减速简洁高效，S 型过渡自然减震，避免冲击。通过合理的规划选择，可有效提升设备精度、效率与性能。深入探索其核心，为工业自动化设备的控制赋能。

ZMC600E 是广州致远电子股份有限公司开发的最新一代高性能智能总线型控制器，是面向工厂智能化时代的机器人控制器。ZMC600E 采用多核异构应用处理器为核心，其内核包括 2 个 64 位的 Arm®Cortex®-A53 核，主频 1.0GHz；4 个 Cortex®-R5F 内核，主频 800MHz。同时板载 1GB DDR4、4GB eMMC 以及 32KB FRAM。

ZMC600E EtherCAT 主站控制器为了满足不同的自动化应用需求，在硬件接口上面，精心设计了 1 路 EtherCAT 主站接口，ms 周期任务抖动 $\pm 5\mu s$ 以内；3 路以太网接口，其中 1 路是千兆网口；2 路 RS485；2 路 CAN；32 个用户 I/O，其中 16 路 DI，16 路 DO，输入支持最多 2 路正交编码或两路脉冲计数；1 个 USB3.0 接口；支持 TF 卡；系统电源采用高稳定隔离电源，支持掉电检测；提供多种程序加密手段，保护用户应用软件知识产权。

ZMC600E 示意图及接口图如下所示：



图1 产品示意图

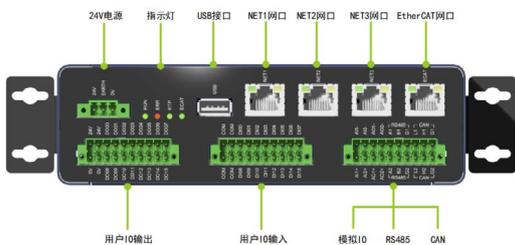


图2 ZMC600E接口图

ZMC600E EtherCAT 主站控制器为设备制造商提供多轴数、多 IO 点控制的新一代可靠智能的运动控制解决方案，提供 T 型、S 型速度规划算法，可以广泛应用于注塑行业、冲压行业、车床行业、搬运码垛、关节机器人、喷涂、玻璃机、压铸机、包装设备、3C 设备、锂电池设备、纺织、流水线

工作站、非标自动化装备、特种机床等高端设备应用。

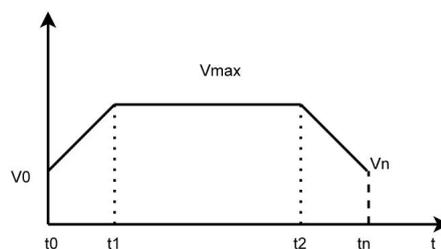
速度规划的概念

速度规划这里特指，在已知运动的起点和终点和完整路径的情况下，基于给定的时间约束，最大速度约束和最大加速度减速度约束的条件，设计运动的速度随时间的曲线。速度规划是运动控制领域中的一个关键环节，其核心目的是为机械设备或机器人在运动路径上设计合适的速度曲线，以确保整个运动过程的平滑性、效率、安全性和舒适性。合理的速度规划能够有效提升设备的运行效率、精度和稳定性，同时减少能耗和机械磨损。因此，在自动化系统中，速度规划是提高性能和可靠性的核心技术之一。常用的速度规划主要为 T 型和 S 型速度规划算法。

T型规划

T 型规划，也称为梯形速度规划，是工业界最广泛采用的形式，它是一种时间最优的简单速度规划方法。它通常包括三个阶段：加速、匀速和减速。在加速阶段，速度从零逐渐增加到设定的最大速度；在匀速阶段，速度保持恒定；在减速阶段，速度逐渐减小到零。

T 型速度规划图示如下：



a)速度曲线

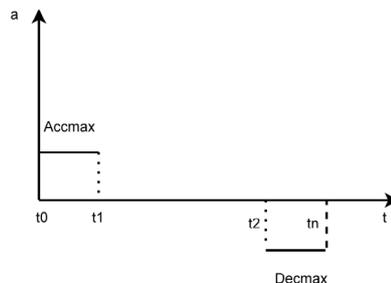


图3 T型速度规划曲线

T 型速度规划问题可以定义为：已知路程 S ，初始速度 V_0 ，结束速度 V_n ，限制条件最大速度 V_{max} ，最大加速度 Acc_{max} ，最大减速度 Dec_{max} 。被控对象从 A 点运动到 B 点，要求生成的速度轨迹在这些条件下时间最优。

从图3可以明显看出，T型规划的加速度表现出不连续性。具体而言，在加速阶段与匀速阶段以及减速阶段与匀速阶段的衔接处，存在着较为显著的突变情况。这种突变会对执行器产生冲击，影响舒适性。T型规划适用于机械加工快速定位及对响应速度要求高、舒适性要求低的工业自动化领域。

S型规划

S型规划，也称为S曲线速度规划，是一种平滑的速度规划方法。S型规划通常包括7个阶段：加加速、匀加速、减加速、匀速、加减速、匀减速、减减速。这种规划方法可以减少机器的振动和噪音，提高运动的平滑性。

S型速度规划图示如下：

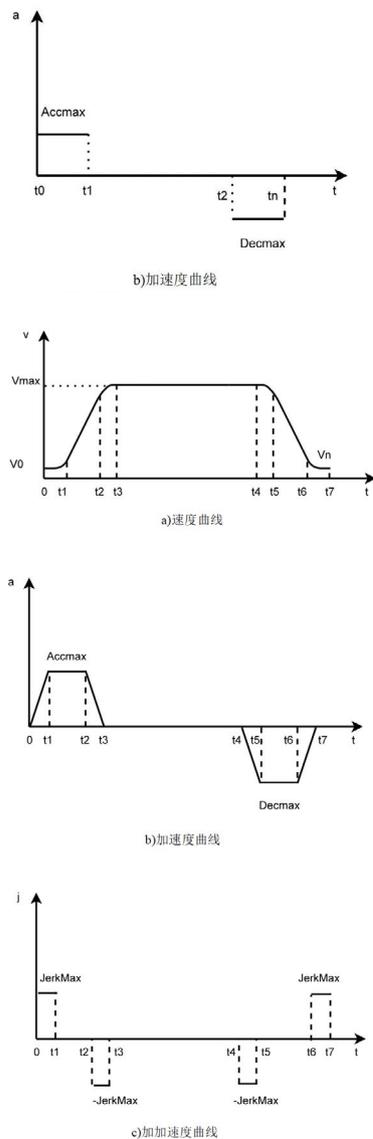


图4 S型速度规划曲线

S型速度规划问题可以定义为：已知路程 S ，初始速度 V_0 、结束速度 V_n 、以及限制条件最大速度 V_{max} ，最大加速度 Acc_{max} ，最大减速度 Dec_{max} 、最大加加速度 $Jerk_{max}$ 。被控对象从起点 A 移动到终点 B ，要求生成的速度轨迹在满足上述约束条件下时间最优。

与 T 型速度规划不同，S 型速度规划在加速和减速阶段不仅考虑加速度的限制，还引入了加加速度 (jerk) 的限制。这使得速度曲线在加速和减速时呈现平滑的 S 型曲线，避免突然的加速度变化，从而减少冲击，提高运动的平稳性。因此，S 型速度规划通常用于对平滑度要求较高的运动控制场景，如机器人、数控机床等应用中。

速度规划API

ZMC600E 运动控制器提供了 T 型和 S 型运动控制 API 接口，方便用户进行速度规划的设置和调整。

创建T型速度规划

指令原型	velocity_handle velocity_new_t(void);
详细说明	创建 T 型速度规划句柄
参数解释	void
返回值	返回创建的速度句柄

创建S型速度规划

指令原型	velocity_handle velocity_new_s(void);
详细说明	创建 S 型速度规划句柄
参数解释	void
返回值	返回创建的速度句柄

删除速度规划的句柄

指令原型	void velocity_delete(velocity_handle handle);
详细说明	删除速度规划的句柄
参数解释	handle: 速度规划句柄
返回值	void

配置速度规划限制参数

指令原型	void velocity_set_config(velocity_handle handle, double v_max, double acc_max, double dec_max, double jerk_max)
详细说明	配置速度规划的相关参数
参数解释	handle: 速度规划句柄 v_max: 最大限制速度 acc_max: 最大加速度 dec_max: 最大减速度 jerk_max: 最大加加速度
返回值	void

设置速度规划

指令原型	int velocity_set_plan(velocity_handle handle, double pos_start, double pos_end, double v_start, double v_end)
详细说明	设置速度规划, 用于建立时间与位置的关系, 这是一个独立的规划, 时间以0开始。
参数解释	handle: 速度规划句柄
	pos_start: 起始位置
	pos_end: 结束位置
	v_start: 初始化速度
返回值	v_end: 结束速度
	返回 0 为规划成功, -1 为输入参数不合理、规划失败

获取执行速度规划的所需的总时间

指令原型	double velocity_get_plan_time(velocity_handle handle);
详细说明	获取执行速度规划的所需的总时间
参数解释	handle: 速度规划句柄
返回值	返回规划的总时间

获取某个时间点的规划信息

指令原型	void velocity_get_plan_slice(velocity_handle handle, double slice, double *pos, double *vel, double *acc);
详细说明	通过时间来获取当前速度规划的切片, 返回该点瞬时的位置、速度、加速度
参数解释	handle: 速度规划句柄
	slice 输入参数, 切片点, 范围应为: 0~规划总时间, 当输入超限时, 会自动限制在范围内
	pos 输出参数, 位置, 可为空
	vel 输出参数, 速度, 可为空
返回值	acc 输出参数, 加速度, 可为空
	void

示例

从 A 点到 B 点路程共 50m, 从起点开始运动, 起始和结束速度为 0, 最大速度 5m/s, 最大加速度 2m/s², 最大减速度 2m/s², 分别使用 T 型和 S 型速度规划, 使用 S 型时最大加加速度 4m/s³。

代码如下:

```
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "velocity.h"

int main(int argc, char *argv[])
{
    if (argc != 2)
    {
        printf("Usage: ./velocity_test 0/1\n");
        return -1;
    }
}
```

```
velocity_handle handle = NULL;

if(strcmp(argv[1], "0") == 0)
{
    handle = velocity_new_t();
    velocity_set_config(handle, 5, 2, 2, 0);
}
else if(strcmp(argv[1], "1") == 0)
{
    handle = velocity_new_s();
    velocity_set_config(handle, 5, 2, 2, 4);
}
else
{
    printf("Usage: ./velocity_test 0/1\n");
    return -1;
}

velocity_set_plan(handle, 50, 0, 0);

double time = velocity_get_plan_time(handle);

for(double t = 0; t < time; t += 0.001)
{
    double pos = 0, vel = 0, acc = 0;
    velocity_get_plan_slice(handle, t, &pos, &vel, &acc);
    printf("pos:%f, vel:%f, acc:%f\n", pos, vel, acc);
}

velocity_delete(handle);

return 0;
}
```

执行结果如下图所示。

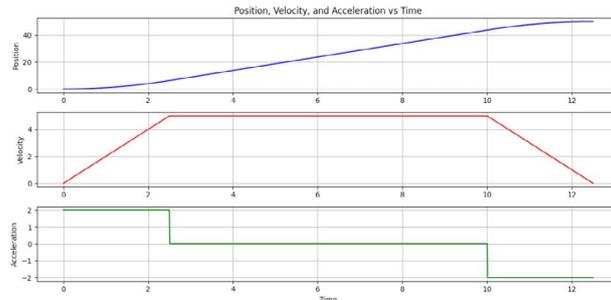


图5 T型速度规划示例

边缘计算 ▼

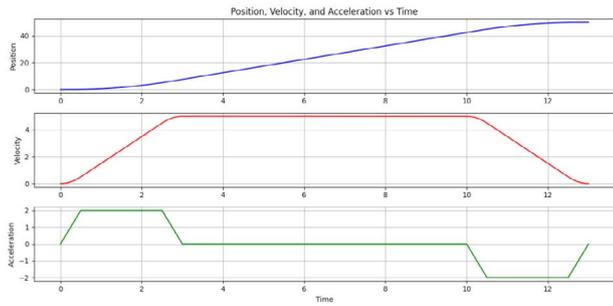


图6 S型速度规划示例

ZMC600E 运动控制器提供 T 型和 S 型速度规划算法可以满足不同应用场景下的速度控制需求。深入掌握速度规划技术，将有助于提高设备的控制精度和效率，为工业生产带来更大的价值。

如需了解更多产品详情，可填写申请表单，
我们会有专人与您联系。

[点击申请](#)

【CAN总线知识】 为何总线“镰刀”波形频频发生？

ZLG 致远电子 2024-12-02 11:31:38

无论是 CAN 总线还是 485 总线，实际应用中经常会出现各种异常，常因总线组网后，波形边沿出现过缓、呈“镰刀”状的现象，导致数据丢失或出错，那么这现象前因后果大家是否真正的了解呢？

案例一

1. CAN总线异常现象

我司某工业机器人客户反馈，使用 SM1500 的机器人控制板卡，在传输数据过程中出现丢帧的情况，如下图 1，客户现场模拟的组网方式为 31 个节点的手拉手拓扑，通讯波特率为 250kbps。



图1 现场组网环境

若总线收发器在使用过程中出现异常，一般会先从总线波形着手去分析原因。如图 2，为客户组网的简要框图，我司使用 CAN 分析仪抓取了第 31 个节点处总线波形，发现波形边沿过缓，出现了“镰刀”状的现象，如下图 3。

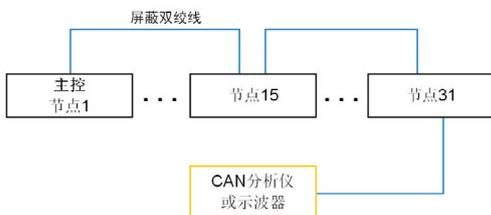


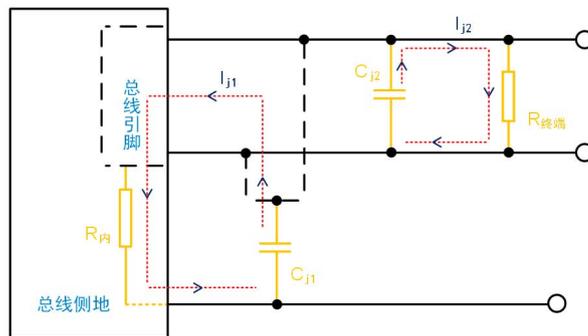
图2 控制板卡组网简要框图



图3 CAN总线“镰刀”波形

总线波形出现“镰刀”状的现象通常是由于总线上存在过大电容起的，根据电容的充放电时间公式可知 $t=RC$ ，其中 R 可看成总线接口内阻与终端电阻， C 则是总线上的等效电容。

如图 4，总线等效电容 C_j 包括总线引脚对地电容 C_{j1} 与总线之间的电容 C_{j2} ，当总线电平由高变低时（压差变化），由于电容上的电压不能突变，那么电容 C_j 会分别通过内阻 R 内和终端电阻 R 终端放电。收发器内阻和终端电阻一般固定，当电容过大时，则放电时间变长，从而导致了总线波形边沿变缓。



SM系列总线隔离收发器

图4 总线等效电容放电原理框图

2. CAN接口电路原理与异常分析

SM1500 CAN 接口电容一般只有几皮法，即使 31 个节点组网最多也不过上百皮法，配合终端电阻使用一般不会出现“镰刀”状波形。我司在检查客户 CAN 接口电路后发现存在 TVS 管、气体放电管等保护器件，如下图 5。TVS 管本身存在较大的结电容，一般在几百到上千皮法，当总线组网后结电容会计增加，高速通讯的时候总线就有可能出现“镰刀”状波形。

互联互通

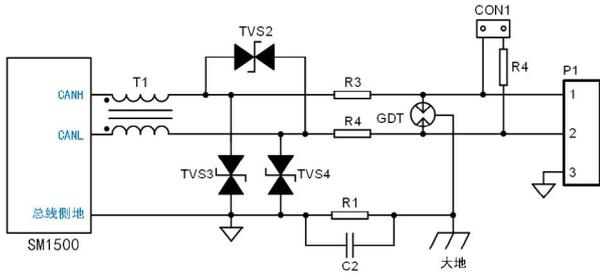


图5 控制板卡CAN接口保护电路

将总线接口保护电路的TVS3和TVS4去掉后组网，并测试第31个节点处波形发现仍呈“镰刀”状，但波形边沿迟缓程度减小，如图6，同时也没有再出现丢帧情况。最后再去掉TVS2后测试，“镰刀”状波形消失，如图7。对比去掉TVS管前后波形，边沿时间由1.3us减小至160ns，如图8。



CAN/CANFD转CAN/CANFD网桥

[点击购买](#)

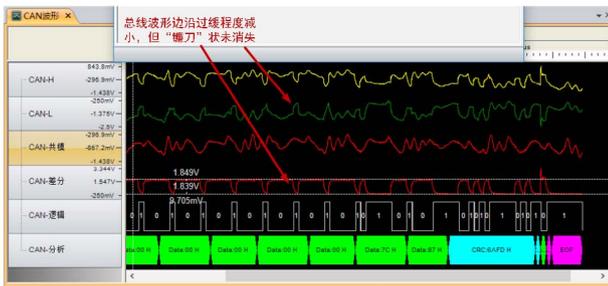


图6 去掉部分TVS管后总线波形



图7 去掉全部TVS管后总线波形

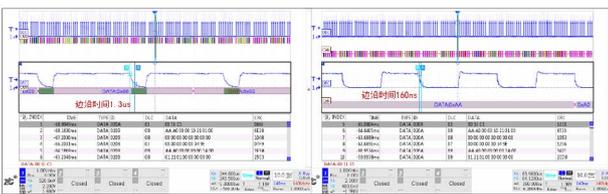


图8 去掉TVS管前后波形边沿时间对比



【热电偶测温】 揭开热电偶滤波电路的神秘面纱

ZLG 致远电子 2024-09-24 11:31:38

在使用热电偶测温模块或设计热电偶测温电路中，每个采集通道往往需要一阶甚至多阶的无源滤波电路，关于热电偶测温无源滤波电路的设计需要考虑什么因素呢？

热电偶测温原理简述

两种不同材料的导体 A、B 与采样电路串接成一个闭合回路，当热端 1 和冷端 2 处于不同的温度 T、T₀ 时，回路中就会产生热电动势 E_{AB}（T、T₀）被 ADC 采集到。当热端和冷端的温度差发生变化时热电动势 E_{AB}（T、T₀）随之变化，用户可根据采集到的电压值查找热电偶温度对照表得出热端 T 与冷端 T₀ 的温度差值，进而得到热端温度值。

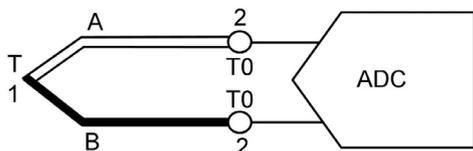


图1 热电偶测温电路简图

输入信号调理

在热电偶测温电路设计中，信号的调理是非常关键的。由于混叠效应，一般 ADC 采集电路前端都需要某些频段的滤波来减少输入噪声，以使模块具备更高的测温精度。

作为一个示例，考虑设计一个截止频率小于工频 50Hz 的低通滤波器电路，另外考虑到采样电压的建立时间，串联电阻值与滤波电容值不能太大，一般设定串联电阻值不高于 500Ω。根据一般经验差模电容值比共模电容值大 10 倍，设计出如图 2 所示一阶低通滤波电路，其截止频率 $f_c = 1 / [2\pi(R_3 + R_4)(C_3 + C_2/2)]$ 。根据该电路图 3 的波特图仿真结果可知其截止频率约为 48.85Hz。

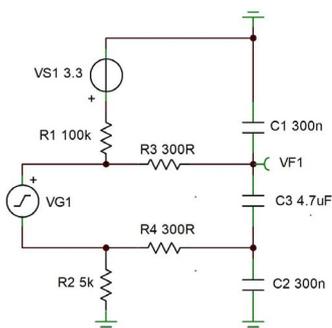


图2 一阶低通滤波电路图

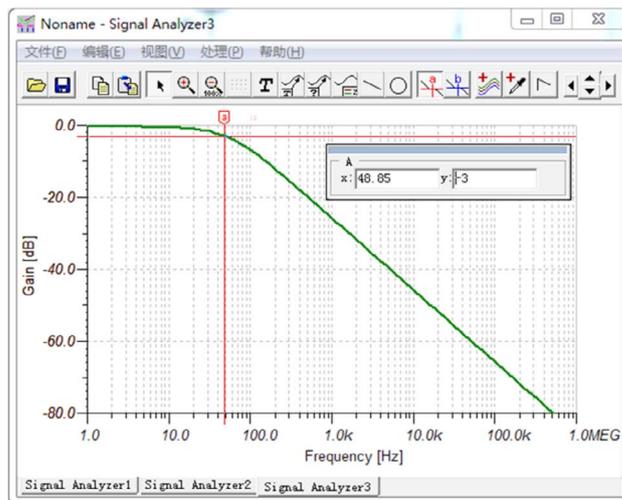


图3 一阶低通滤波电路波特图

根据工程设计经验，考虑到电阻的温漂及电容的压电效应等因素影响，另外考虑到设计余量我们一般会截止频率设计的尽可能低一些。例如可能将截止频率设计在 25Hz 左右，此时 C₃ 的容值可能接近 8μF。考虑到大电容的成本以及一阶滤波的滚降不够理想，因此我们采用二阶滤波的方式进行改善。

我们可以在图 2 的电路基础上再增加一阶电路改为无源二阶低通滤波器如图 4 所示，根据该电路图 5 中的波特图仿真结果，可知该电路截止频率的仿真值为 24.56Hz。

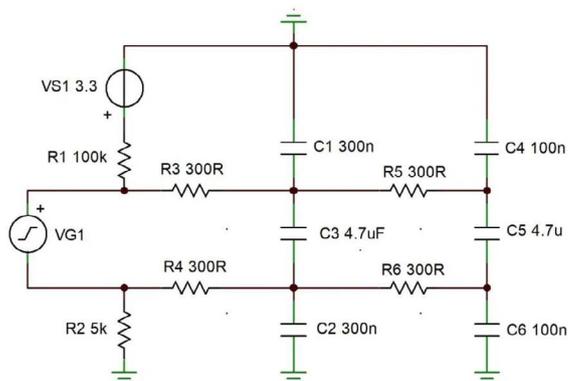


图4 二阶低通滤波电路图

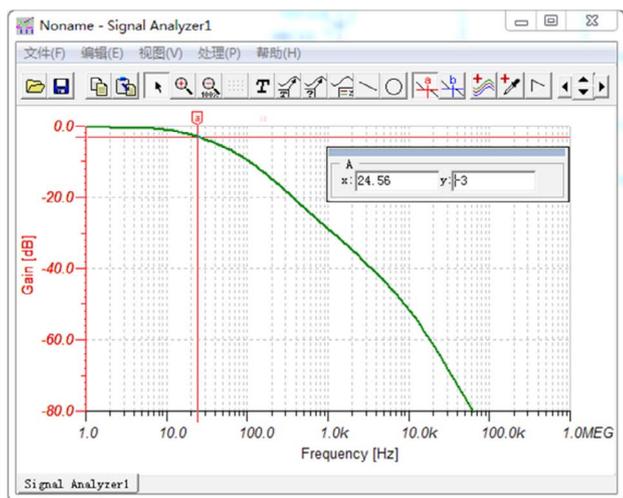


图5 二阶低通滤波电路波特图

总结

热电偶测温低通滤波电路设计的截止频率越低，对高频噪声的抑制能力越强，但是截止频率越低阻容值要求越大采样电压建立时间越长。截止频率设置过高，对工频或其它频段干扰的衰减不够理想会对测温精度产生影响。因此热电偶测温低通滤波电路截止频率设计的过高或过低对测试结果均会产生影响，根据经验值截止频率可设计在 30Hz~40Hz 左右。

热电偶测温模块 ZAM6218A 的推荐滤波电路采用二阶滤波截止频率约为 40Hz，该产品具备优异的采样精度和测温范围，通过对采集数据进行处理后 ZAM6218A 通过 I2C 协议直接输出热端温度数据，大大简化了用户的软件和硬件设计，是热电偶测温的不二选择。



图6 ZAM6218A简易说明图

2024/12 第12期

微文摘

ZLG MICRO DIGEST



ZLG致远电子官方微信