

文章编号: 2096-1618(2023)06-0643-06

开源固件 Marlin 的 LabVIEW 驱动程序及应用

唐智超¹, 李仲豪², 蒋世奇¹, 张杰³, 郑富中¹

(1. 成都信息工程大学自动化学院, 四川 成都 610225; 2. 四川轻化工大学机械工程学院, 四川 自贡 643000; 3. 重庆电子工程职业学院智能制造与汽车学院, 重庆 401331)

摘要:针对 Marlin 开源 3D 打印机固件无法用 LabVIEW 开发应用程序, 不能利用已有硬件和 LabVIEW 的便捷性快速构建适合用户需要的 3D 控制平台或数据采集系统, 设计一款驱动程序。该驱动程序利用串口助手和示波器的瞬间波形抓取功能, 识别 Marlin 的 USB 端口与外部硬件的通信机制, 并结合 LabVIEW 中串口工具的虚拟仪器软件架构(VISA)控件, 构建能与 Marlin 串行通信的串口读取子 VI 与初始化控件, 形成稳定的 LabVIEW 驱动 Marlin 程序, 最后成功应用于固件的温度控制与结果显示。该方案提高了 Marlin 的应用灵活性, 节省了二次开发成本, 为 3D 控制平台及数据采集系统的构建提供有效参考。

关键词: Marlin; LabVIEW; 驱动程序; 3D 控制; 数据采集

中图分类号: TP216

文献标志码: A

doi: 10.16836/j.cnki.jcuit.2023.06.004

0 引言

Marlin 是一款开源 3D 打印机固件, 其强大之处在于支持 X-Y-Z 直角结构, coreXY、SCARA、三角洲等多种不同结构的 3D 打印机, 支持多种硬件电路板与多种计算机语言, 还附加了自动调平等功能。正因为 Marlin 在 3D 打印系统中的强大功能且运行稳定, 各种基于个人计算机的 3D 打印软件也层出不穷, 如: Reptier-Host、replicatorG 和 SelfCAD 等, 基于 C++ 的微型液晶控制面板也已相当成熟。但这些技术不能与其他设备或仪器取得有效衔接和信息互通, 不利于 3D 控制平台的二次开发, 缺乏灵活性^[1]。

LabVIEW 是图形化编程语言, 随着 PC 能力的增强而不断发展其功能, 使面向对象编程、多线程、ActiveX, 以及在此基础上发展起来的消息队列框架(QMH)、队列消息(JKI)状态机、异步消息通信(AMC)等多线程 LabVIEW 程序框架应运而生, 进一步简化了科研和工程技术人员的工作^[2-5]。这些技术可以快速实现 Marlin 与其他设备或仪器的灵活重组和互联互通, 同时还可以把 Marlin 的各个硬件功能模块便捷地整合成具有独立功能的 LabVIEW 控件或子虚拟仪器程序文件(virtual instrument file, VI), 供其他 LabVIEW 应用程序调用^[6]。

近年来, 关于通用串口的 LabVIEW 程序应用已十

分广泛, 但是针对开源固件 Marlin 的 LabVIEW 驱动程序还没有出现, 不能有效利用 Marlin 固件上 3D 电机驱动、温度控制与获取、限位开关的状态获取、风扇的速度控制等功能。因此, 如何利用 LabVIEW 编制 Marlin 的驱动程序, 迅速完成 3D 控制平台与其他工程仪器或科研设备的信息互通, 为基于高速总线或宽带网络的 LabVIEW 集散控制系统的构建提供低成本技术实现的新途径。

1 Marlin 的串口解码

驱动程序的开发步骤依次为: 了解 Marlin 的硬件结构, 串行字符串的识别, 串口读取子 VI 的实现, 驱动程序初始化和温度控制与结果显示应用, 如图 1 所示。

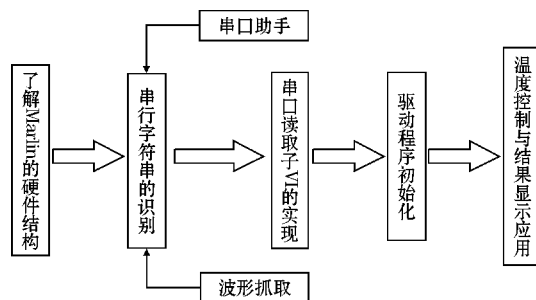


图 1 开发步骤框图

1.1 Marlin 的硬件结构介绍

Marlin 始于 2011 年, 经过 10 余年的发展, 目前最新的版本为 2.0, 固件上配置的内容主要包括: 1 个

收稿日期: 2022-10-17

基金项目: 四川省科技厅重点资助项目(2021YFG0147); 重庆市教委科技计划资助项目(KJZD-K202103101)

通信作者: 郑富中. E-mail: zfz@cuit.edu.cn

STM32 7805 主控芯片、1 个 CH340G 数据传输转换芯片、1 个 USB 通信接口、4 个 42 步进电机的驱动接口、2 个 12 V 直流电源输出接口、另配 2 组加热电源、X-Y-Z 3 个轴的限位开关以及 2 个温度传感器等,如图 2 所示。具有利用 PID 控制加热温度、温度检测精度高、支持圆弧轨迹的规划、在中断中处理步进电机的梯形加减速、限位开关触发时有中止运作的功能等。

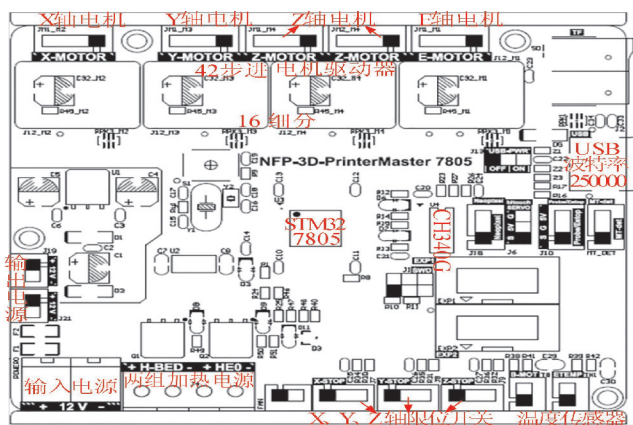


图 2 Marlin 的硬件结构组成图

1.2 串口字符串的识别

由 Marlin 的硬件组成可知,固件上配有通用串行总线(universal serial bus,USB)接口,能实现与 PC 的快速通信,相当于测控系统中的下位机^[7-9]。为节省资源,固件上的微处理器 STM32 7805 仍采用异步收发传输器(universal asynchronous receiver/transmitter, UART)与外设进行数据收发,而实现 USB 与 UART 转换的芯片是 CH340G,如图 3 所示^[10]。

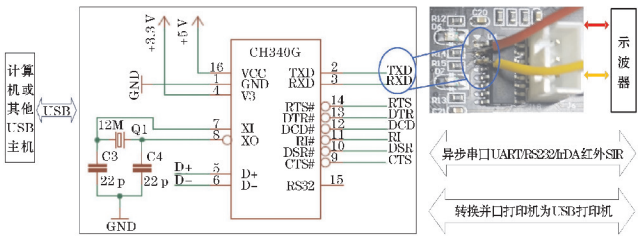


图 3 CH340 的工作方式示意图

在串口方式下,CH340 提供常用的调制解调器联络信号,用于计算机扩展异步串口;或者将普通的串口设备直接升级到 USB 总线,芯片内置 USB 上拉电阻,它的 D+ 与 D- 引脚直接连到 USB 的两条总线上;又或者直接外加红外收发器即可构成 USB 红外线适配器,实现 SIR 红外线通信。在并口方式下,可转换并口打印机为 USB 打印机。

要实现 Marlin 的驱动编程,先在 PC 上安装 CH340G 的芯片驱动程序,建立 PC 与 Marlin 的通信。

为确认通信是否已建立,利用串口助手软件 Termite 3.4 检测串口通信是否正常。方法是先在 PC 的设备管理器中确认串口编号后,再打开串口助手选中此编号并设置波特率为 250000。在串口助手软件的输入窗口中发送 G-code 命令 M119,获取限位开关状态,在显示区中就会显示出收发的消息,并报告 3 个限位开关的状态,如图 4 所示。当 3 个限位开关完全处于断开状态时,显示区将显示第一条 M119 命令获得的消息;当把图 2 右下部的 Y 轴限位开关用金属线短路后再发送 M119 命令,显示区将获得图 4 中第二条 M119 命令的应答消息,说明 PC 与 Marlin 已建立正确的通信。

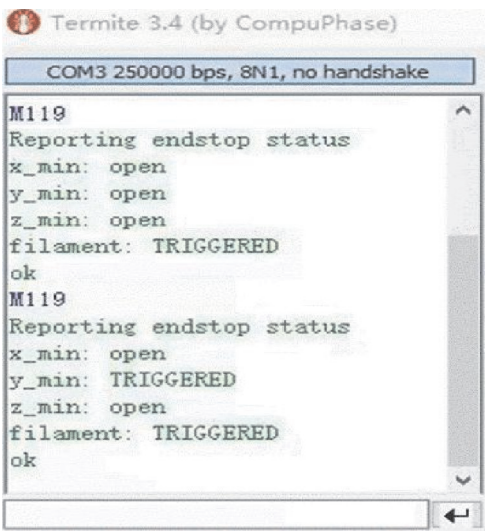


图 4 串口助手检测通信状态

为更准确地掌握消息的传输方式,将 CH340G 的 2 和 3 引脚分别与示波器的两个信号输入端相连,如图 3 的右上角所示,上面的引线接示波器的 1 号通道,产生的信号显示在示波器屏幕的下半部分,下面的引线接示波器的 2 号通道,产生的信号显示在示波器屏幕的上半部分^[11]。

当 PC 通过串口助手发送 M119 命令时,通过抓取瞬间波形,示波器屏幕下半部分就会出现一串高低电平的方波,时序方向是从右向左,信号下的二进制数从左向右每 10 位的前两位 1 和 0 表示一个字符的停止位和起始位。当信号有了一个从 1 到 0 的跳变表示发送开始,UART 传送的数据是由低位到高位,需要把二进制数作反转,每 8 位逐位对调后,产生第二排二进制数,再查 ASCII 码对照表,由此可知 PC 向 Marlin 发送了 M119 指令与换行符。同理,示波器屏幕上半部分显示的接收到的消息头为 Rep……(图 5)。图 6 接收到的消息尾经过译码为 ok 与换行符,与图 4 接收到的消息头与消息尾是一致的。通过以上实验,确立了 PC 与 Marlin 的数据收发方式。

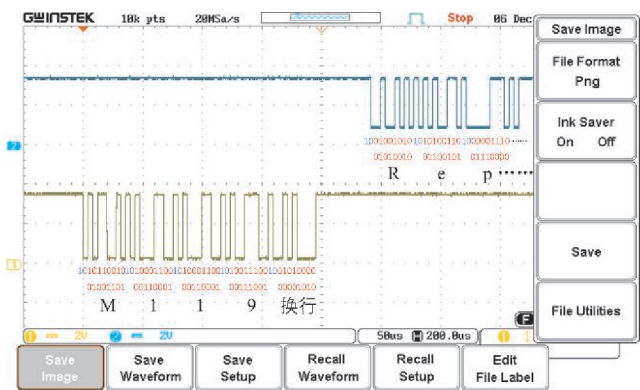


图 5 命令字符串与消息头字符串

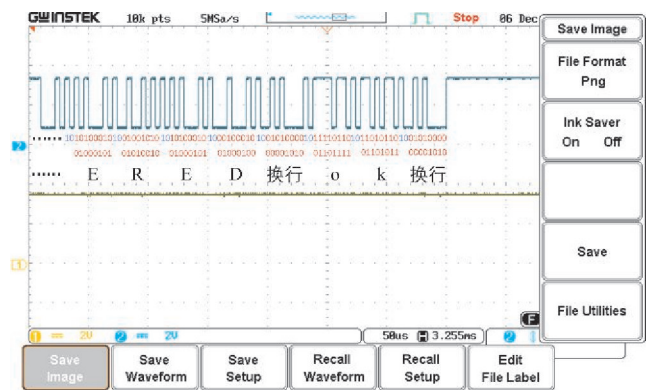


图 6 消息尾字符串

2 Marlin 的 LabVIEW 驱动程序

利用串口助手或具有瞬间波形抓取功能的示波器,获取Marlin的响应字符串,再利用此字符串消息

尾和 LabVIEW 中的虚拟仪器软件架构(virtual instruments software architecture,VISA)控件,可快速建立串行读取子 VI 和初始化控件。串口读取程序流程图 7 所示,其中的双横线代表多线程并行。

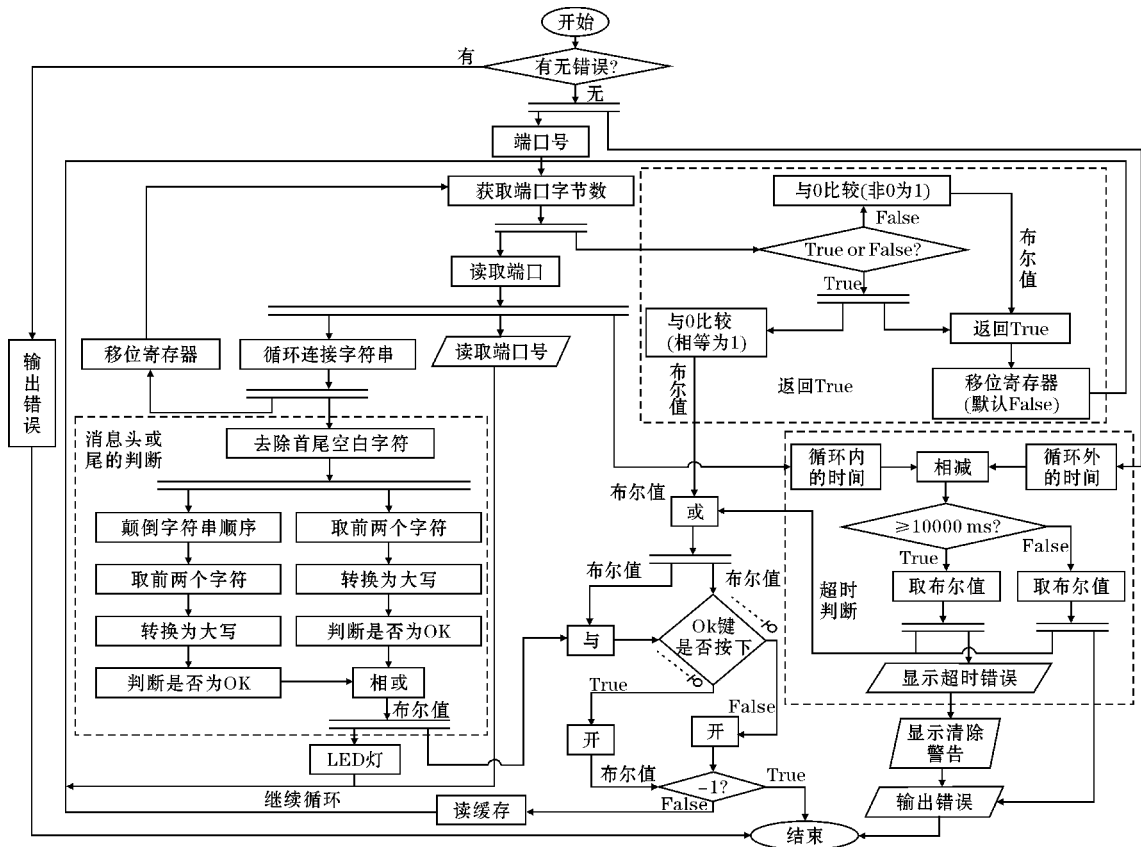


图 7 串口读取程序流程

2.1 串口读取子 VI 的实现

程序开始后,由条件结构检查错误输入控件中是否有错误产生。若有错误,直接输出错误后结束;若没有错误,由设备输入控件输入设备端口号,并发送至“Instr”属性节点。字节属性获取端口字节数并输出端口号后,同时进行以下 3 个任务:消息头或尾的判断,返

回 True 和超时判断。
任务 1,消息头或尾的判断。其步骤:先将从端口读取到的字符串循环连接后送入“While 循环”的移位寄存器,等待循环结束或继续从端口读取字符串,同时将连接好的字符串去除首尾空白字符。接着,判断前两个字符的大写是否为 OK,或颠倒字符串后判断前两个字符的大写是否为 OK,将以上两个 OK 判断结果

做逻辑或运算,即判断字符串头或尾是否存在 ok 字符。若存在则逻辑或控件输出布尔值 1,点亮 LED 灯;否则逻辑或控件输出布尔值 0,关闭 LED 灯。同时,逻辑或控件输出的布尔值作为“消息头或尾的判断”任务的输出待用。当任务 2 和任务 3 的布尔值相或后,再和任务 1 的待用输出相与布尔值作为是否关闭“While 循环”的依据。

任务 2,返回 True。其步骤:先判断“While 循环”的移位寄存器中存储的布尔常量是“True”还是“False”,默认为“False”,判断“Instr”属性节点输出的字节数是否不为 0,若为 0,则移位寄存器返回“False”,断续等待,直到“Instr”属性节点输出的字节数不为 0,则移位寄存器返回“True”,进入另一个条件,此条件保持移位寄存器中的布尔值为“True”,且“Wait for OK”控件为“False”。“While 循环”是否结束由任务 2 和任务 3 的输出布尔值相或后决定,判断“Instr”属性节点输出的字节数是否为 0。若不为 0,则由任务 3“超时判断”决定是否结束“While 循环”。如果没有超时,则继续循环,直到“Instr”属性节点输出的字节数为 0。然后结束“While 循环”,停止串口读取。最后输出端口号,读取缓存,LED 灯保持熄灭。如果“Wait for OK”控件被置为“True”,当“Instr”属性节点输出的字节数为 0,任务 2 和任务 3 的布尔值相或后输出必为 1,则等待字符串中的 ok 出现。当任务 1 的布尔值输出为 1 时,LED 灯被点亮,则可结束循环,停止串口读取,最后输出端口号,读取缓存。

任务 3,超时判断。其步骤:用循环外的时间减去循环内的时间,其中,循环内的时间是指程序每次在“While 循环”内进行字符串读取时,循环内的一个时钟开始计时的时刻,循环外的时间是指进入“While 循环”的时刻,两者的差值与设定的默认等待 10 s 比较。若大于等于设定时间则报错,并输出布尔值 1。如果“Wait for OK”控件被置为“False”,则结束“While 循环”;如果“Wait for OK”控件被置为“True”,则等待 ok 字符串来结束“While 循环”^[12]。若小于设定时间,同样分“Wait for OK”控件为“True”和“False”两种情况讨论,但是否结束循环则由任务 1 和任务 2 的输出布尔值决定。

2.2 驱动程序初始化

要调用串口,首先要对串口进行初始化编程,整个初始化程序也可被封装为一个控件,简称初始化控件,其程序流程如图 8 所示。以默认方式打开 VISA 控件,先通过“VISA 资源名”的枚举型控件选择准备好的端口号,如果作为控件则可通过创建引脚来接收输入^[13]。通过 VISA 清空源缓存,判断有没有错误产生。

若有错误,关闭资源,清除错误;同时,通过“Instr”属性节点获得设备的初始化接口类型,与预先选定的接口类型比较,产生真假两种条件,并设定等待获取的超时时间为 10 s。如果获取的接口类型与预先在枚举型字符串常量控件中选定的字符串 Serial 不相等,进入假条件,重新选取对应的接口类型;如果相等,则进入真条件,进行串口设置。在串口设置部分,首先给端口属性赋值,通过“Instr”属性节点将输入与输出的结束终端字符都设定为 0 换行符,波特率设置为 250000。然后清空接收与发送的缓存,再将接收和发送的缓存大小都设置为 4096 字节。随后,进行条件判断“有无错误?”。若有,则直接输出错误和端口号;若无,则进入假条件中,调用串口读取子 VI 进行串行读取,延时 3 s 且无需字符串首或尾的 ok 字符串来停止串行读取,防止没有消息。然后向对应的串口缓存中写 M27 指令加换行符字符串,清除未知消息,再次调用串口读取子 VI,超时参数采用默认 10 s 且需要字符串首或尾的 ok 字符串来停止读串口,结束后输出端口号和错误,供其他控件使用,同时生成初始化控件。

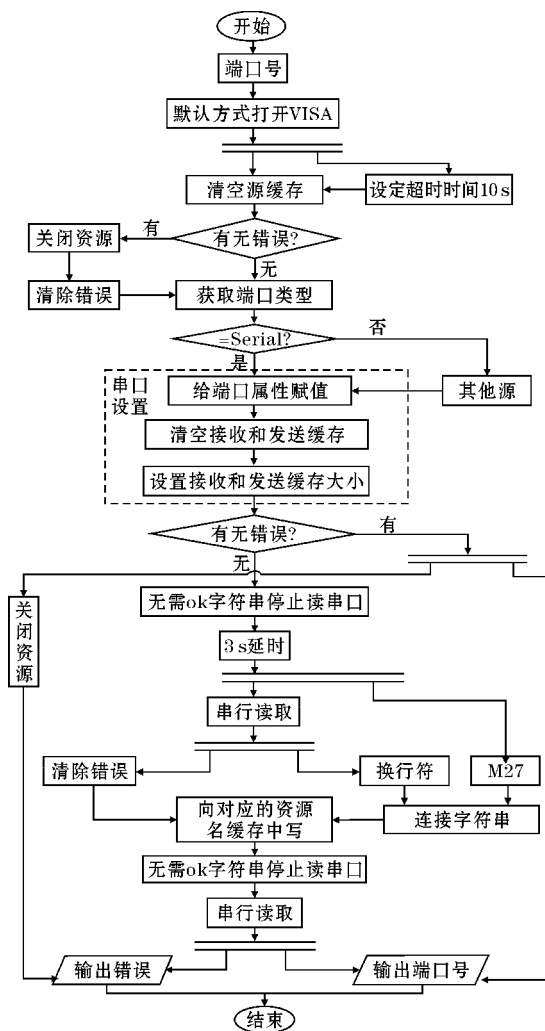


图8 初始化程序流程图

3 实验结果与分析

初始化程序可以作为公共的函数控件出现在 Instrument I/O 下的 Instr Drivers 面板中供其他程序调用,而串口读取程序则是作为私有的子 VI,可通过复制或拖拽的方式调用,进一步还可建立 Marlin 上各硬件的 VI 控件^[14-15]。以 Marlin 对 3D 打印机中热床和打印头的温度控制与结果显示为例,演示 Marlin 串口驱动的运行效果。串口完成初始化后,设置打印头温度为 200 °C,热床温度为 60 °C,如图 9 所示。程序设置约 10 ms 从串口读取一个温度数据,并把图形的更新频率设置为 1 s,确保实时清楚地观察到曲线的变化^[16-18]。从图 9 可以看出,约在 80 s 时,打印头的温度达到预设的 200 °C。

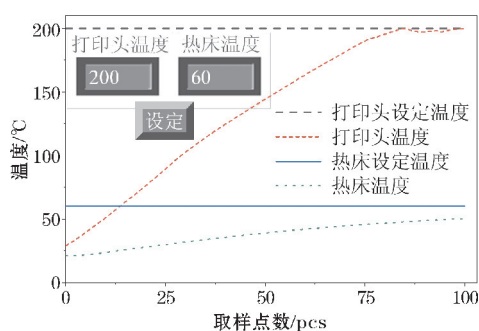


图9 初设温度跟踪曲线

接下来将打印头和热床的温度又分别设置为 100 °C 和 50 °C,经过约 1 s 的时间即调整为设定值。随后打印头的温度立刻开始下降,而热床的温度仍有一个较短的上升过程才开始缓慢下降,如图 10 所示,这与两次温度设置的温差有关。当温差设得较大时,温度变化会较明显,温差设得较小时温度变化较缓慢。

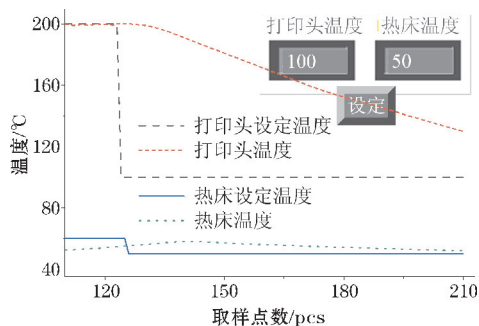


图10 变温跟踪曲线

4 结束语

为实现开源固件 Marlin 的 LabVIEW 串口驱动程

序,利用串口助手和示波器的瞬间波形抓取功能,准确地掌握 Marlin 通过 USB 接口的消息发送与接收方法,从而确立在 PC 上利用 LabVIEW 编程驱动 Marlin 的串行通信模式,开发串行读取子 VI 和初始化控件,以供开发 Marlin 的其他硬件功能时调用。在此基础上,以温度控制与结果显示为例,调用初始化控件和串行读取子 VI,同时利用 LabVIEW 的图形化显示控件,对 Marlin 的两个温度传感器通路实验了实时的温度控制与温度数据采集及其图形化显示。为 3D 运动控制设备的开发和优化提供了便利,具有良好的应用前景。

实验结果表明针对 Marlin 的 LabVIEW 驱动程序能够稳定运行,完成了 PC 与 Marlin 之间的串行数据通信,为 Marlin 的高效利用开辟了新路线。

致谢:感谢过程装备控制工程四川省高校重点实验室资助项目(GK202010)对本文的资助

参考文献:

- [1] 何颖,潘盛湖,鲁豪杰,等. 基于运动控制器的 3D 打印机控制系统设计[J]. 机械工程与技术, 2018,7(4): 285-289.
- [2] Kodosky J. LabVIEW [M]. Proc. ACM Program. Lang. 4, HOPL, Article 78, 2020:54.
- [3] Bitter R, Mohiuddin T, Nawrocki M. LabVIEW™ Advanced programming techniques [M]. Taylor & Francis Group, CRC press, Computer Science, Engineering & Technology, 2017.
- [4] Zhai X, Zhang M, Bai X, et al. A signal analysis design based on LabVIEW [C]. 2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery. IEEE, 2015:473-476.
- [5] Guan Y, Shen B, Zhang Y, et al. Design of color mixing 3D printing system based on LabVIEW [J]. J Comput, 2017, 28(6): 277-287.
- [6] Wang W B, Li J Y, Wu Q J. The design of a chemical virtual instrument based on LabVIEW for determining temperatures and pressures [J]. Journal of Automated Methods and Management in Chemistry, 2007, 68:143.
- [7] Qiu Y, Chen Q. The realization of cyclic serial port communication based on LabVIEW [C]. 2014 International Conference on System Science and Engineering (ICSSE). IEEE, 2014:46-49.
- [8] Machacek J, Drapela J. Control of serial port (RS-232) communication in LabVIEW [C]. 2008 In-

- ternational Conference-Modern Technique and Technologies. IEEE,2008:36-40.
- [9] 马敏,杨胜伟. 基于 LabVIEW 的气体监测系统串口通讯设计[J]. 自动化与仪表,2016,31(11):43-46.
- [10] Asha K R,Tasleem P S,Kumar A V R,et al. Real time speed control of a DC motor by temperature variation using LabVIEW and Arduino[C]. 2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT). IEEE,2017:72-75.
- [11] He F. Measurement and analysis of transient information characteristics of IOT equipment using graphical programming[C]. Journal of Physics: Conference Series. IOP Publishing, 2021, 1982(1):12156.
- [12] 曼世超,卢会国,卢勇. 基于 LabVIEW 的心电信号检测平台的设计[J]. 成都信息工程大学学报,2016,31(3):255-260.
- [13] 吴苗,许江宁,缪绣华,等. 基于 LabVIEW 的设备驱动程序开发技术研究[J]. 微计算机信息,2006(S2):153-155.
- [14] Hongfu Z,Xinyan X,Yong T. Serial Communication Interface Design Based on LabVIEW and VC Mix Programming[C]. 2007 8th International Conference on Electronic Measurement and Instruments. IEEE,2007:344-349.
- [15] Cheng H X,Zhang K,Cheng L. Design a temperature control system based on LabVIEW[C]. Applied Mechanics and Materials. Trans Tech Publications Ltd,2014,644:95-98.
- [16] Zhu W,Liu J,Yang H,et al. Design of high precision temperature measurement system based on LabVIEW[J]. Int. J. Adv. Comput. Sci. Appl., 2015,6:153-155.
- [17] 朱炜敏,刘瑾,杨海马. 高精度恒温控制系统的设计[J]. 测控技术,2017(4):89-92.
- [18] 许奇,刘瑾,杨海马. 基于 PLC 与 LabVIEW 的恒温控制系统设计[J]. 测控技术,2016,35(9):55-58.

LabVIEW Driver and Application of Open-Source Firmware Marlin

TANG Zhichao¹, LI Zhonghao², JIANG Shiqi¹, ZHANG Jie³, ZHENG Fuzhong¹

(1. College of Automation, Chengdu University of Information Technology, Chengdu 610225, China; 2. College of Mechanical Engineering, Sichuan University of Science and Engineering, Zigong 643000, China; 3. Intelligent Manufacturing and Automobile School, Chongqing College of Electronic Engineering, Chongqing 401331, China)

Abstract: As LabVIEW is unable to develop applications for Marlin open-source 3D printer firmware, the convenience of existing hardware and LabVIEW could not be used to quickly build a 3D control platform or data acquisition system to meet users' needs. This driver program used the serial port assistant and the oscilloscope's instantaneous waveform capture function to identify the communication mechanism between Marlin's USB port and external hardware, and combined the Virtual Instrument Software Architecture (VISA) control of the serial port tool in LabVIEW to build a serial port reading sub VI and initialization control that could communicate with Marlin serially, forming a stable LabVIEW driven Marlin program. Finally, the driver program is successfully applied to the firmware's temperature control and result display. This project improves the application flexibility of Marlin, saves the secondary development cost, and provides an effective reference for the construction of 3D control platform and data acquisition system.

Keywords: Marlin; LabVIEW; driver program; 3D control; data acquisition