DOI:10.20033/j.1003-7241.(2026)07-0071-05

面向一体化关节的驱控软件更新技术研发

张成林[1,2], 夏科睿[2], 党珊珊[2]
(1.江苏大学 管理学院,江苏 镇江 212013; 2.合肥哈工轩辕智能科技有限公司,合肥 230000)

摘　要:一体化关节是协作机器人以及人形机器人的关键核心零部件。针对一体化关节的模块化协作机器人的伺服驱控软件在线更新技术,是基于 MODBUS 通讯协议,设计地址区域划分,将 Flash 区域划分为两块完全独立空间地址区域,每个地址区域交替执行软件更新,保证备份软件的有效性与功能性。设计 Bootloader 引导程序,将引导程序刷写进控制器,将驱控软件伺服板与电脑端的上位机进行通信连接,实现伺服驱控应用软件的在线更新与备份。搭建软硬件验证平台,开发可视化上位机软件执行应用程序数据解析与下发,编写验证程序以及设计一致性校验。通过软件下发 Bootloader 引导程序,引导程序下发至应用程序起始地址,完成软件更新。实验分析表明,设计的 Bootloader 能正确引导加载程序的运行,制定的软件一致性校验与软件备份方案,保障了软件更新的稳定性与可靠性, 解决 ECU 的功能异常问题,提高了一体化关节软件研发和应用调试过程的工作效率。

关键词:软件更新;软件备份;一体化关节;机器人;伺服系统;引导程序;一致性校验
中图分类号: TP242;TP311.1　　**文献标志码**: A　　**文章编号**: 1003-7241(2026)07-0071-05

Research on updating technology of integrated joint drive and control software

Zhang Chenglin[1,2], Xia Kerui[2], Dang Shanshan[2]
(1. School of Management, Jiangsu University, Zhenjiang 212013, Jiangsu, China;
2. Hefei Hagong Xuanyuan Intelligent Technology Co., Ltd., Hefei 230000, China)

Abstract: Integrated joints are key core components of collaborative robots and humanoid robots. This paper presents an online update technology for the servo drive-control software of modular collaborative robots based on integrated joints. Based on the MODBUS communication protocol and address partitioning design, the Flash memory is divided into two completely independent address spaces. These spaces perform software updates alternately, ensuring the validity and functionality of the backup software, a Bootloader is designed and flashed into the controller. By establishing communication between the servo drive-control board and the PC-based host computer, online updates and backups of the application software are realized. A hardware-software verification platform is built, and a visual host computer software is developed to execute data parsing and downloading. Verification procedures and consistency checks are also designed. The update process involves downloading the Bootloader, which then directs the write operation to the application′s start address to complete the update. Experimental analysis demonstrates that the designed Bootloader correctly guides the program loading. The proposed consistency check and backup schemes ensure the stability and reliability of software updates, resolve functional anomalies in the ECU, and significantly improve the work efficiency of the R&D and debugging processes for integrated joint software.

Keywords: software update; software backup; integrated joint; robot; servo system; bootloader; consis-tency check

随着工业对制造业技术水平要求的不断提高,机器人智能技术得到各个领域的广泛应用[1-3],而机电控一体化关节是机器人的核心组成单元和重要研究方向。目前,驱控一体机器人伺服驱动软件功能需求不断完善,软件迭代速率逐步加快。伺服系统驱控一体化是机器人伺服关节的必然趋势[4],在伺服驱动器的研发及调试使用过程中,需要对主控芯片进行多次固件烧录、升级。

基于串口实现软件升级的方式[5-6],由于人为因素或外界因素的不确定性,在擦除系统原有代码后,可能出现软件更新失败或者更新软件本身代码逻辑问题,导致更新软件无法运行、设备无法继续投入使用等风险。

为防止出现问题时电子控制单元(electronic control unit,ECU)的基本功能失效,文献[7]提出数字信号处理器(digital signal processor,DSP)在线升级容错性能的方案,但设备因程序问题导致无法正常工作风险依然存在;文献[8]提出了一种基于 TMS320F28335 通过 GPIO 引脚控制现场可编程门阵列(field programmable gate array,FPGA)软件在线升级方法,主要适配 DSP+FPGA 架构;文献[9]通过 GATT 通用属性协议实现两个连接设备的数据通信实现方法,但缺乏应用程序备份策略;为防止车载

收稿日期:2025-03-12;**录用日期**:2025-05-30
基金项目:安徽省科技重大专项(202003a0502015)
作者简介:张成林(1989—),男,硕士研究生,工程师,研究方向为机器人、智能装备与机电一体化技术等。
引用本文:张成林, 夏科睿, 党珊珊. 面向一体化关节的驱控软件更新技术研发[J]. 自动化技术与应用, 2026,45(7):71-75. (Zhang Chenglin, Xia Kerui, Dang Shanshan. Research on updating technology of integrated joint drive and control software[J]. Techniques of Automation and Applications, 2026, 45(7):71-75.)

ECU功能失效,一般车载ECU采用基于统一诊断服务(unified diagnostic services,UDS)诊断协议实现软件数据备份刷写策略,通过引导程序恢复上一版本软件和数据的功能[10-11],但车载UDS诊断协议并不适用于机器人伺服系统软件。

本文提出了基于机器人伺服系统通用工业通讯总线MODBUS协议的双分区软件升级技术,通过Bootloader引导程序进行双分区软件升级管理,实现软件升级失效自动恢复上一版软件或停留在引导程序手动执行软件更新的功能,提高了ECU工作的稳定性与可靠性。

1 一体化关节的驱控系统

1.1 一体化关节的结构

为实现机器人的模块化,一体化机器人关节需集成电机、减速机、编码器、制动器、驱动器等部件,形成独立的驱动单元[12],如图1所示,为满足不同关节的合理搭配,机器人关节需具备系列化,可实现不同的扭矩配比。

1.2 软件更新与备份设计

伺服驱控软件更新与备份系统设计主要包含上位机与底层引导程序两个主要模块:上位机软件模块主要执行应用软件程序代码编译输出数据二进制(binary,BIN)文件的解析与数据下发,MODBUS通讯协议指令下发以及当前运行软件的版本确认。

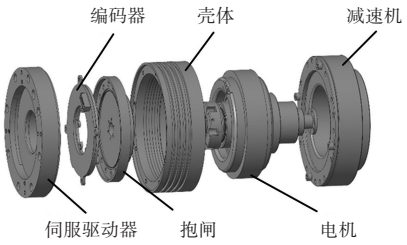


图1 一体化关节结构

Fig. 1 Structural of the integrated joint

底层引导程序模块主要通过MOD BUS协议与上位机建立通信,将待更新伺服驱动软件下载到DSP芯片预设地址区间,检测所述伺服驱动软件的数据下载是否完整有效,更新伺服驱动软件应用程序如果有效则运行最新应用程序,如果更新伺服驱动软件无效则回滚到上一版本应用程序,DSP芯片下电复位,检测所述伺服驱动软件当前运行版本是否为设定软件版本等功能。软件更新与备份方案执行的流程图如图2所示。

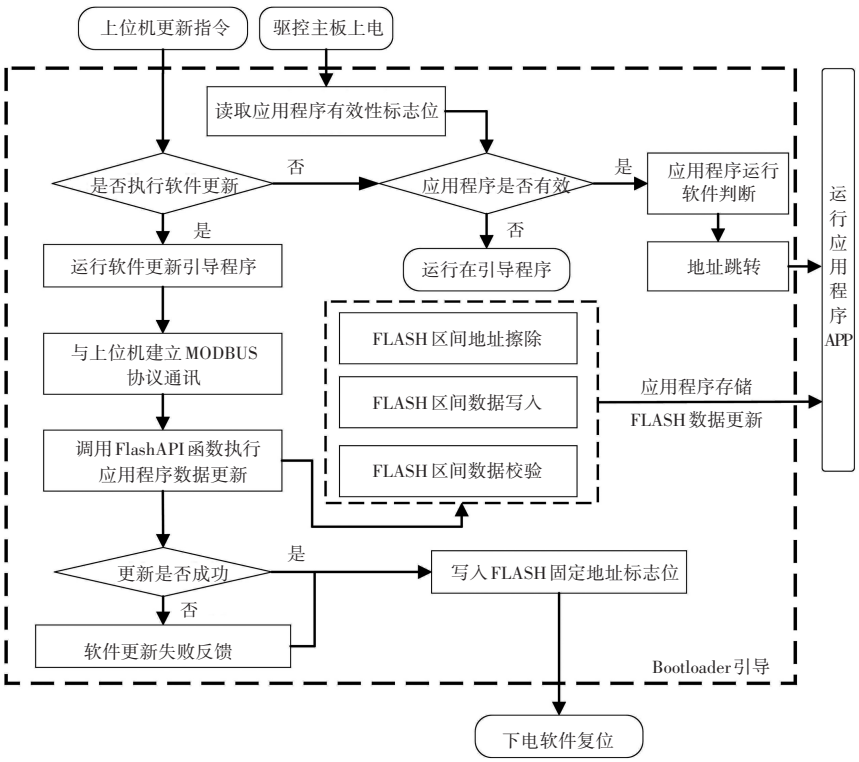


图2 软件在线更新方案执行流程图

Fig. 2 Execution flowchart of the software OTA update solution

2 软件引导程序

2.1 地址区域划分

该方案为了实现异常情况的数据备份升级,该策略要求ECU的应用软件Flash区域容量至少为应用程序的2倍,将Flash区域划分为两块完全独立空间APP1和APP2,APP1地址区域与APP2地址区域交替执行软件更新,保证备份软件的有效性与功能性。本方案则为Bootloader+App1+App2,即App1可以运行应用程序,App2也可以运行应用程序,在Bootloader启动的时候,判断当前是运行App1还是App2,然后跳转至相应的区域运行[13-14]。

DSP芯片预设地址区间指待更新应用程序APP1(APP2)地址区间与机械臂伺服上次软件更新执行的驱动

软件烧写应用程序 APP2(APP1)地址区间,包含地址区间的分配方式:

1) 若机械臂伺服控制主板 DSP 芯片上次软件更新执行的驱动软件烧写应用程序 APP2(APP1)地址区间为地址区间 Addr2(Addr1),则待更新应用程序 APP1(APP2)地址区间为 Addr1(Addr2)。

2) 若机械臂伺服控制主板 DSP 芯片上次软件更新执行的驱动软件烧写应用程序 APP2(APP1)地址区间为地址区间 Addr1(Addr2),则待更新应用程序 APP1(APP2)地址区间为 Addr2(Addr1)。

3) 地址划分的判断方式通过在每次烧写结束固定地址内写入判断标识位。

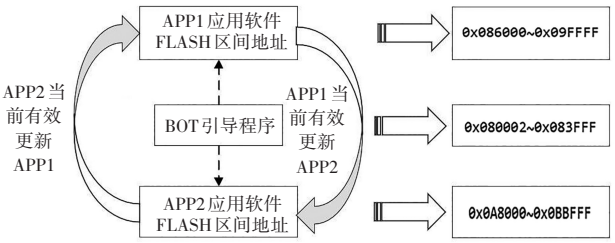


图 3 软件地址划分区域

Fig. 3 Software address allocation layout

2.2 地址区域划分

协作机械臂伺服驱动软件每次执行 DSP 芯片上下电,首先在引导程序内读取烧入固定 Flash 地址内写入的判断标识位,判断标识位包含 DSP 芯片地址区间 Addr2(Addr1)是否包含有效应用程序 APP1(APP2)与是否执行协作机械臂伺服驱动软件更新操作。软件引导流程判断逻辑包含:

1) 是否包含有效应用程序 APP1(APP2)判断标志位,若没有地址区间 Addr1(Addr2)内具备有效应用程序 APP1(APP2),并且,是否执行协作机械臂伺服驱动软件更新操作为否或是都停留在 DSP 芯片引导程序。

2) 是否包含有效应用程序 APP1(APP2)判断标志位,若只有一个地址区间内 Addr1(Addr2)具备有效应用程序 APP1(APP2),并且,是否执行协作机械臂伺服驱动软件更新操作,是则直接跳转到有效应用程序 APP1(APP2),否则停留在引导程序等待上位机执行软件更新操作。

3) 是否包含有效应用程序 APP1(APP2)判断标志位,若有两个地址区间内 Addr1 与地址区间 Addr2 都具备有效应用程序 APP1 与有效应用程序 APP2,并且,是否执行协作机械臂伺服驱动软件更新操作为否,是则直接跳转到最近一次更新的协作机械臂伺服驱动软件应用程序 APP1(APP2),否则停留在引导程序等待上位机执行软件更新操作。

2.3 引导程序设计

引导程序 Bootloader 代码逻辑[15],如图 5 所示。其中,主要有通过初始化 InitSys Ctrl()函数对系统硬件外设时钟进行初始化;通过 InitFlash()函数初始化内部 Flash。主循环 while()内周期性调用 FL_ MainFunction()函数执行 Fl_Erase()擦除、Fl_ProgramPage()写入以及 FL_CheckSum ing()数据一致性校验[16],状态转换顺序为擦除→数据写入→校验,状态转换不能跳转,逐级进行。

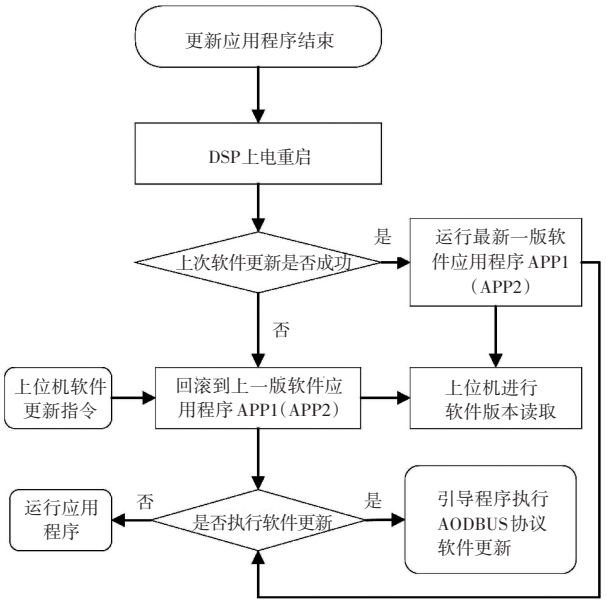


图 4 软件上电引导执行流程

Fig. 4 Software power-onbootloader execution flowchart

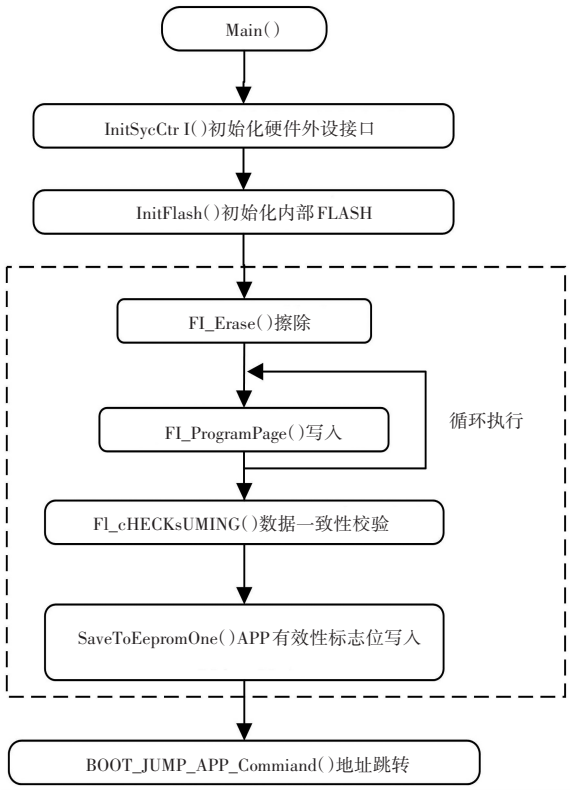


图 5 软件数据下载引导程序代码逻辑

Fig. 5 Code logic of software data download bootloader program

2.4 软件更新上位机

上位机基于 MODBUS 协议通信执行数据解析与数据传输。上位机软件具备不同的数据文件解析函数,用于解析嵌入式系统编程软件(code composer studio,CCS)生成

的数据格式S-Record、HEX以及BIN文件。如图6所示，用户根据加载的不同格式的数据文件，上位机调用相应的文件解析函数。其中，S-Record、HEX格式文件包含数据地址，数据内容、数据段地址以及校验值等信息；BIN格式文件内只包含数据信息，数据解析内容需要上位机根据已知的下位机FLASH数据段地址分配进行数据长度与地址配置。

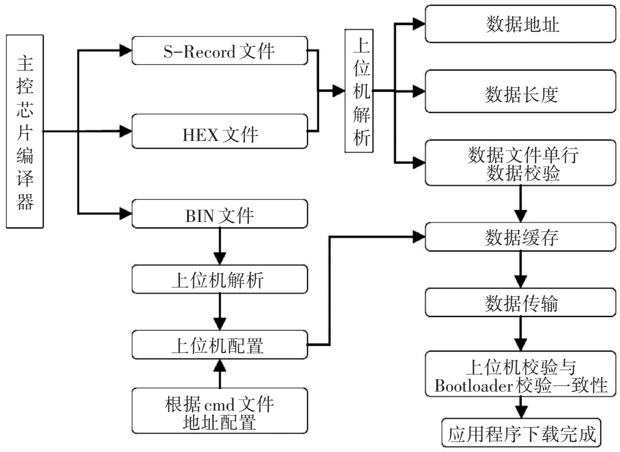


图6 上位机数据解析与下载流程

Fig. 6 Flowchart of host computer data parsing and downloading

3 实验验证

3.1 环境搭建

搭建测试环境，对本次设计的备份刷写方案进行测试。应用程序根据使用实际需求进行代码编写，需要调用CCS编译工具将应用程序.out文件生成.BIN文件，给控制器ECU上电，将BootLoader引导程序刷写进控制器中，然后通过通信设备将驱控软件伺服板与电脑端的上位机进行通信连接，完成应用软件的更新。软硬件搭建环境图7所示。

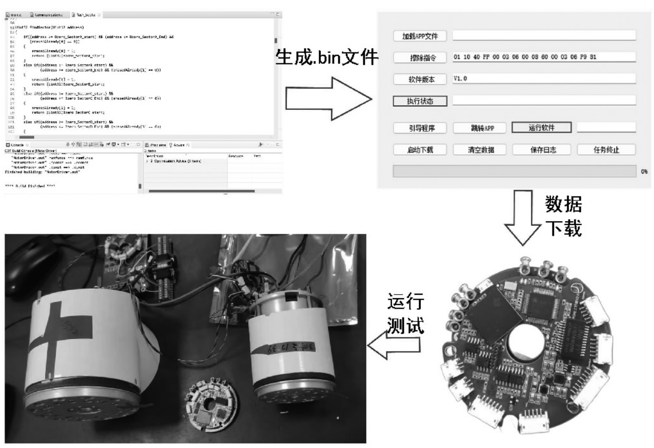


图7 软件在线更新方案执行流程

Fig. 7 Flowchart ofthe software online update scheme execution

3.2 一致性校验

验证平台搭建完成后开始试验验证数据解析校验是否具备一致性，如表1所示。

表1 数据解析校验一致性统计表

Tab. 1 Consistency statistics for data parsing and verification

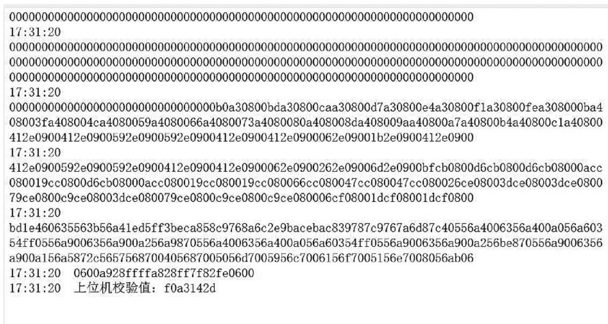
序号	数据类别	APP1	APP2
1	软件版本	AV.01.01	BV.01.01
2	解析数据长度	0x2290E	0x2290E
3	起始地址	0x86000	0xA8000
4	BIN文件首地址值	0x0049	0x004B
5	ECU刷新首地址值	0x0049	0x004B
7	上位机计算校验值	0xF0A3142D	0x76D2CF74
8	ECU计算校验值	0xF0A3142D	0x76D2CF74

验证操作步骤如下：

1）通过CCS软件编写应用程序代码，根据应用程序地址不同生成两个BIN文件，导入上位机软件，数据解析，通过32位校验算法对所有解析数据进行校验，生成数据校验值。

2）驱动控制板上电，上位机与控制器通讯连接，下发软件跳转执行bootloader引导程序，跳转成功发送擦除指令，控制板收到指令后执行擦除，成功返回正确的响应，失败返回负响应。

3）待接收到Flash擦除成功消息后，启动文件烧写过程。待烧写完成，执行数据下载校验，校验一致。刷新软件有效标志位写入外部存储eeprom内，引导程序跳转至应用程序起始地址，应用程序成功运行，表明软件刷新成功，如图8所示。



（a）上位机下载显示校验值



（b）上位机可视化软件刷新

图8 可视化上位机一致性校验

Fig. 8 Consistency check of the visual host computer

3.3 软件更新验证

正常工作状态下，伺服驱动软件需要具备软件更新功能，但是当软件更新过程中已下发擦除指令将该地址的软件擦除，下载数据更新过程突然中断，执行软件一致性校验失败，更新新版本软件内存溢出等问题，软件备份功能可以让伺服驱动软件回到之前版本的软件，提高了ECU

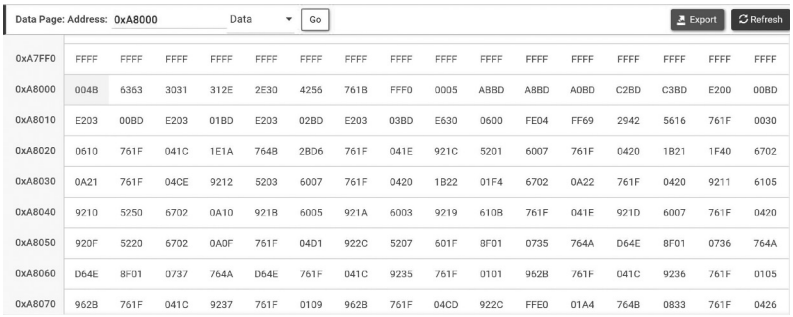
工作的稳定性与可靠性。

实验测试软件下载过程中断,更新失败后,ECU 下电重启,上电读取执行软件更新成功的标志位,标志位失败,引导程序根据判断程序有效性跳转至之前的软件版本,不影响当前伺服驱控软件的正常运行。如图 9 所示,通过编译工具读取两个 APP 应用软件对应存储地址 Flash 区内数据,可以得到地址 0x86002-0x86005 存储数据解析 ACCII 码为软件版本号“AV. 01. 01”,地址 0xA8002-0xA8005 存储数据解析 ACCII 码为软件版本号“BV. 01. 01”,证明软件的有效性。



0x85FF0	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF
0x86000	0049	4363	3031	312E	2E30	4156	761B	FFF0	0005	ABBD	A8BD	A0BD	C2BD	C3BD	E200	00BD
0x86010	E203	00BD	E203	01BD	E203	02BD	E203	03BD	E630	0600	FE04	FF69	2942	5616	761F	0030
0x86020	0610	761F	041C	1E1A	7649	0BD6	761F	041E	921C	5201	6007	761F	0420	1B21	1F40	6702
0x86030	0A21	761F	04CE	9212	5203	6007	761F	0420	1B22	01F4	6702	0A22	761F	0420	9211	6105
0x86040	9210	5250	6702	0A10	921B	6005	921A	6003	9219	610B	761F	041E	921D	6007	761F	0420
0x86050	920F	5220	6702	0A0F	761F	04D1	922C	5207	601F	8F01	0735	7648	B64E	8F01	0736	7648
0x86060	B64E	8F01	0737	7648	B64E	761F	041C	9235	761F	0101	962B	761F	041C	9236	761F	0105
0x86070	962B	761F	041C	9237	761F	0109	962B	761F	04CD	922C	FFE0	01A4	7648	E833	761F	0426

(a) 应用软件 APP1 地址 addr1 数据



0xA7FF0	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF
0xA8000	004B	6363	3031	312E	2E30	4256	761B	FFF0	0005	ABBD	A8BD	A0BD	C2BD	C3BD	E200	00BD
0xA8010	E203	00BD	E203	01BD	E203	02BD	E203	03BD	E630	0600	FE04	FF69	2942	5616	761F	0030
0xA8020	0610	761F	041C	1E1A	764B	2BD6	761F	041E	921C	5201	6007	761F	0420	1B21	1F40	6702
0xA8030	0A21	761F	04CE	9212	5203	6007	761F	0420	1B22	01F4	6702	0A22	761F	0420	9211	6105
0xA8040	9210	5250	6702	0A10	921B	6005	921A	6003	9219	610B	761F	041E	921D	6007	761F	0420
0xA8050	920F	5220	6702	0A0F	761F	04D1	922C	5207	601F	8F01	0735	764A	D64E	8F01	0736	764A
0xA8060	D64E	8F01	0737	764A	D64E	761F	041C	9235	761F	0101	962B	761F	041C	9236	761F	0105
0xA8070	962B	761F	041C	9237	761F	0109	962B	761F	04CD	922C	FFE0	01A4	764B	0833	761F	0426

(b) 应用软件 APP2 地址 addr2 数据

图 9　驱控单元 ECU 软件更新

Fig. 9　Drive-control unit ECU software update

4　结论

本文通过伺服系统软件更新技术研究实验验证得出基于机器人一体化关节伺服系统,设计的软件可视化上位机具备数据分析与软件更新一致性校验功能,操作简单;基于软件更新实验证明本文软件循环更新备份方案实现双分区系统软件循环升级与备份功能,解决 ECU 的功能异常问题,提高了软件的可靠性与稳定性;基于 MODBUS 协议的 Bootlader 引导程序支持系统异常信息的在线读写,有助于软件异常问题的在线分析。针对伺服系统软件升级技术,未来可以融合物联网云-管-边-端一体化技术[17],实现智能化伺服系统双分区软件升级的策略,从系统层面采用远端机器人伺服驱动单元的数据处理与软件升级模式,更高效地实现伺服系统 CPU 内存的资源利用,简化伺服驱动软件参数的标定流程,提高机器人伺服系统的可靠性与灵活性。

参考文献

[1]矫洪成, 吴文凯, 刘禹欣, 等. 机器人技术在智能建造中的应用与发展现状[J]. 自动化技术与应用, 2024, 43(5):1-5, 14.
[2]任焕梅. 机械工业智能化背景下的智能制造系统开发与优化[J]. 自动化技术与应用, 2025, 44(12):168-172.
[3]赵丹. 基于机器学习的轮式机器人关节爬坡运动轨迹控制方法[J]. 自动化技术与应用, 2025, 44(1):36-40.
[4]焦露, 张秋菊, 陈宵燕, 等. 机器人一体化关节模组驱控系统设计[J]. 现代制造工程, 2023(9):54-60.
[5] 陈艺芃, 林飞浪, 郭大鸣, 等. 基于 Bank Swap 的 STM32 软件升级方案[J]. 电子设计工程, 2022, 30(3):144-148.
[6]薛一哲, 吴左富, 王昭. 基于状态机策略的 DSP 串口软件升级方案[J]. 工业控制计算机, 2022, 35(9):27-28.
[7]詹克旭. 基于 UDS 协议的汽车 ECU 升级方案[J]. 计算机应用与软件, 2019, 36(1): 191-196, 203.
[8]陶媛媛, 杜彬, 田彬. 基于车载控制器 BootLoader 的数据备份刷写软件方案实现[J]. 汽车电器, 2022(9):39-41.
[9]吴海帆, 李树东, 朱广亚. 基于 ESP32 的脚踝康复机器人的控制系统设计[J]. 自动化技术与应用, 2025, 44(6):136-139;
[10]周恒, 梁贵友, 柳旭, 等. 基于实时操作系统的车载 ECU 双分区软件空中下载升级技术[J]. 汽车工程学报, 2023, 13(6):803-809.
[11]朱锦新, 范鹏飞, 翟长远, 等. 潮汐式育苗营养液循环灌溉系统设计[J]. 农机化研究, 2025, 47(8):84-91.
[12]金力, 王成军, 夏科睿. 驱控一体化机器人关节的研制及应用[J]. 机床与液压, 2021, 49(21):62-66.
[13]杨朝阳, 黄凯旋, 仝秀峰, 等. 基于 UDS 的 Bootloader 上下位机设计[J]. 软件, 2023, 44(7):42-47.
[14]张砚雪. 基于物联网和卷积神经网络的智能农机安全驾驶系统[J]. 农机化研究, 2025, 47(3):211-216.
[15]罗博, 仇傲. MC9S12X 系列单片机 PE 生成代码的串口 Bootloader 设计[J]. 电子设计工程, 2021, 29(20):138-142.
[16]马建辉, 于良杰, 李研强, 等. Bootloa der 程序 Flash 数据的一致性验证方法[J]. 单片机与嵌入式系统应用, 2018, 18(12):19-21, 52.
[17]代品宣, 黄华, 周院平. 基于“云-管-边-端”物联网架构的网点管控系统的设计[J]. 物联网技术, 2025, 15(10):35-38, 44.