

航天器展开机构压力传感器 无线测控系统设计

汪新舜, 吴星云, 刘晓飞, 邢香园, 吴剑锋

(上海卫星装备研究所, 上海 200240)

摘要: 传统的航天器展开机构任务压力传感器参数测控系统使用有线的方式, 展开过程中存在力轴转动网线跟随缠绕的现象, 以往需根据位置调整采集终端, 避免网线断裂, 使用极不方便, 且单点采集, 不利于多通道部署和数据集成处理; 为了解决上述问题, 开发了一种基于 ESP8266 和 LabVIEW 的嵌入式分布式无线压力传感器参数测控系统, 设计信号调理电路, ADC 采集模块, 使用 ESP8266 无线模块多点分布式组网; 终端采集软件使用 LabVIEW 虚拟仪器平台, 采用多线程低耦合的 QMH 队列框架, 功能模块化编写, 便于开发人员维护, 实现设备通讯, 传感器校准及数据采集处理等功能; 使用 PSP 服务器动态实时共享试验数据, 实现数据的网络快速传递, 提高系统的数据处理速度; 添加了增量式积分分离 PID 算法, 实现了传动机构与传感器的联动, 精确控制压力参数, 同时具备数据动态拟合及超限报警等功能; 系统运行稳定, 数据采集频率和压力控制精度高, 具有工程应用价值。

关键词: 分布式; 无线采集; ESP8266; 虚拟仪器; PID

Design of Wireless Measurement and Control System for Spacecraft Deployment Pressure Sensor

WANG Xinshun, WU Xingyun, LIU Xiaofei, XIN Xiangyuan, WU Jianfeng

(Shanghai Institute of Spacecraft Equipment, Shanghai 200240, China)

Abstract: The pressure sensor parameter acquisition of traditional spacecraft deployment tasks uses the wired connection, during the process of deployment, there is a phenomenon of the deployment mechanism winding the cable around the rotating axis. Previously, it was necessary to adjust the acquisition terminal according to the cable's position to avoid the cable breakage, which has the characteristics of extremely inconvenient, single channel acquisition, inconvenient to multi-channel acquisition and data integration processing. In order to solve the above problems, an embedded and distributed wireless pressure sensor parameter acquisition system based on ESP8266 and LabVIEW is developed, which designs the signal conditioning circuit and ADC acquisition module, the ESP8266 wireless module is used to deploy the multi-point distributed networking. The terminal acquisition software is based on LabVIEW virtual instrument platform, which adopts the multithreading and low coupling QMH queue software framework and function module, it is convenience for the developers to achieve the functions of the equipment communication, sensor calibration, data acquisition and processing. The PSP server is used to dynamically share the test data in real time, realize the rapid data transmission through the network, and improve the data processing speed. The incremental integral separation PID algorithm is added to realize the linkage between the transmission mechanism and the sensor, which accurately controls the pressure parameter and the data dynamic fitting and overrun alarming. The system has the features of the stable operation, high frequency of data acquisition, high control accuracy, and engineering application value.

Keywords: distributed; wireless acquisition; ESP8266; virtual instrument; PID

0 引言

为了模拟航天器电池板在外太空环境中的零重力展开, 需在地面使用动力传动设备模拟太空中的零重力环境^[1], 在试验的过程中, 需实时采集控制压力传感器等数据, 目前国内主要采用在传动轴的顶部安装压片式力传感器, 内部嵌入数据采集板卡, 使用有线网络的方式将数据传输至电脑终端处理。传动轴进行全方位转动时, 存在线缆缠绕的现象,

操作人员需及时调整位置进行试验, 且传感器单点采集, 无法对多个传感器组网, 导致现场布置和试验不便。随着无线物联网技术的迅速发展, 市场推出了一系列的物联网芯片和 IOT 解决方案, 现有的物联平台存在以下问题^[2-5]:

- 模拟量数据采集频率较低, 不能满足试验 50 Hz/S 的采集速率;
- 模拟量数据采集精度较低, 不能满足试验 12 位的采集精度;

收稿日期: 2023-04-06; 修回日期: 2023-06-06。

作者简介: 汪新舜(1990-), 男, 硕士, 工程师。

引用格式: 汪新舜, 吴星云, 刘晓飞, 等. 航天器展开机构压力传感器无线测控系统设计[J]. 计算机测量与控制, 2024, 32(3): 9-16, 29.

- 3) 大多数产品使用数字量信号采集, 采集电压电流信号时的功耗过高, 电池用电快;
- 4) 原有采集软件不能进行自定义的数据处理, 包括数据拟合, 加速度拟合, 数据联合比对等;
- 5) 原有的物联平台采集软件不具备 PID 算法, 无法完成模拟量参数和传动机构的联合控制。

针对以上问题, 根据航天某院航天器机构展开试验任务要求, 研制了一套基于 ESP8266 和 LabVIEW 的嵌入式分布式无线压力传感器参数采集控制系统, 硬件采用 STM32、ADC 及 ESP8266 芯片, 设计了信号调理电路、信号采集电路、信号发送电路及低功耗智能供电管理电路。上位机软件使用 LabVIEW 虚拟仪器平台, 采用多线程低耦合的 QMH 队列框架, 功能模块化编写, 便于开发人员维护, 可实现设备通讯, 传感器零位校准, 数据采集处理等功能, 使用 PSP 服务器动态实时共享试验数据, 实现数据的网络间快速交换, 提高系统的数据处理速度, 同时具备数据动态拟合及超限报警等功能; 在固件程序中采用循环中断的架构, 实现了设备初始化配置, 网络配置连接服务器, 数据采集无线发送, 新增了增量式积分分离 PID 算法, 完成了数据采集和传动机构的联合控制, 解决了试验过程中存在的非线性、参数时变性和模型不确定性等问题, 减少了压力控制的超调量, 增强了控制精度^[6]。

1 测试需求

嵌入式分布式无线压力传感器参数采集系统的硬件参数及要求如表 1 所示。

表 1 测控系统硬件参数

项目	指标要求
采集通道	5 路
力传感器量程	0~50 kg
采集精度	12 位(全量程)
采集频率	30~50 Hz
压力控制精度	±0.5 kg
连续工作时间	100 h
连接方式	无线

软件的功能需求包括: 上位机可实时监测各点支撑力的工作情况, 显示未工作(灰色)、工作中(绿色)及报警(红色)等多种工作状态; 上位机可完成服务器网络配置, 多个无线传感器连接组网, 无线传感器数据采集存储及 PID 压力控制设置参数的下发; 上位机可实时监测各点支撑力的波动, 自动拟合支撑力的波动曲线, 拟合曲线可以和实时曲线在同一坐标系下显示; 上位机可实时监测产品速度的变化趋势, 自动拟合出速度波动曲线, 拟合曲线可以和实时曲线在同一坐标系下显示; 上位机具有力传感器阈值设置过载报警显示功能, 可实时显示力传感器的剩余电量, 使用特定数据格式保存数据; 上位机具备数据回看等功能。

2 系统硬件及原理设计

嵌入式分布式无线采集系统基于 STM32 微处理器和 ESP8266 无线模块设计, 设计 ADC 信号采集, 信号调理及无线信号发送电路。传感器将力物理量线性转换为电流模拟量, 信号调理模块将电流转换为标准的输出电压, ADC 模块采集电压, 将模拟量转换为数字量, 传输至微处理器的串口端, 微处理器使用指令配置无线模块连接路由, 工作在网络客户端模式, 将接收到的传感器数据包发送至无线模块, 无线模块通过连接无线路由组网, 将数据传送至上位机服务器, 其余观测客户端可远程连接 PSP 服务器, 读取共享数据进行处理。无线传感器动态接收上位机的压力参数控制设置值, 与动力传动轴使用串口通讯, 采用增量式积分分离 PID 算法控制传统轴的转动速度来动态调节压力参数, 上位机负责数据的滤波处理显示存储及报警设置等功能, 硬件连接原理图如图 1 所示。

试验人员将多个带有嵌入式无线采集板卡的力传感器安置在试验平台上, 打开电源开关, 嵌入式无线采集控制板卡上电自检, 完成网络配置后, 开始连接无线路由器, 连接上位机服务器, 如果上位机服务器未打开, 连续发送请求等待连接终端服务器指令直至连接成功, 连接无线路由成功 LED 指示灯为蓝色, 不成功为红色, 连接上位机服务器成功指示灯为绿色。试验人员打开采集软件界面, 配

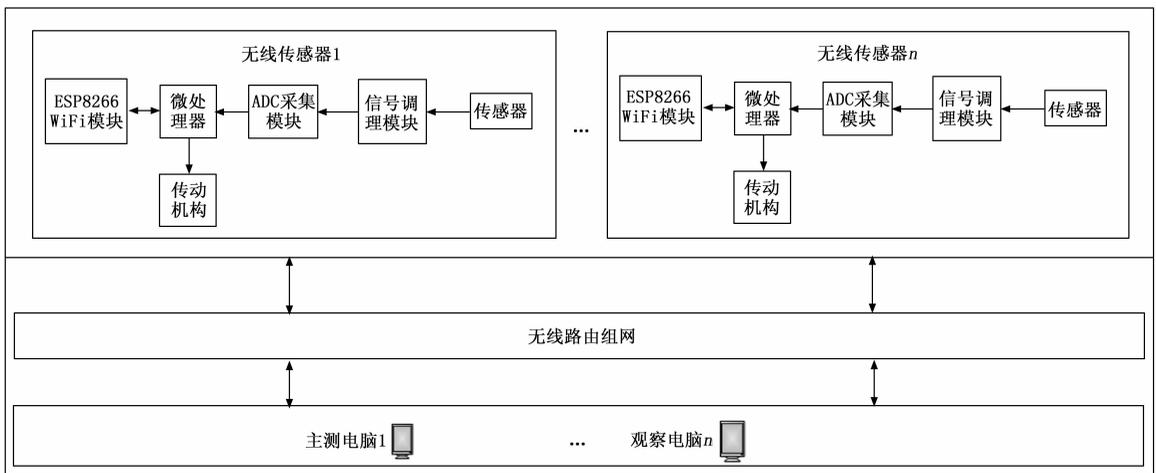


图 1 硬件连接示意图

置 IP 地址, 端口号, 打开服务器, 建立网络连接, 网络连接成功后, 传感器按照软件编写的采集队列依次发送数据, 数据实时显示在界面上, 点击拟合按钮, 界面显示加速度拟合曲线, 试验人员可自由选择传感器数据在同一波形图中做对比查看, 数据实时存入 PSP 服务器, 其他的观测电脑也可以同步读取数据查看, 选定某个传感器, 发送压力设定参数, 进行 PID 控制, 设定传感器参数报警上限值, 一旦数据超限, 软件发出提示, 如果试验过程中, 无线传感器发生通讯故障, 软件报错提示试验人员, 软件尝试重新连接, 重新连接失败后, 及时更换传感器, 重新采集, 继续试验, 系统运行的流程如图 2 所示。

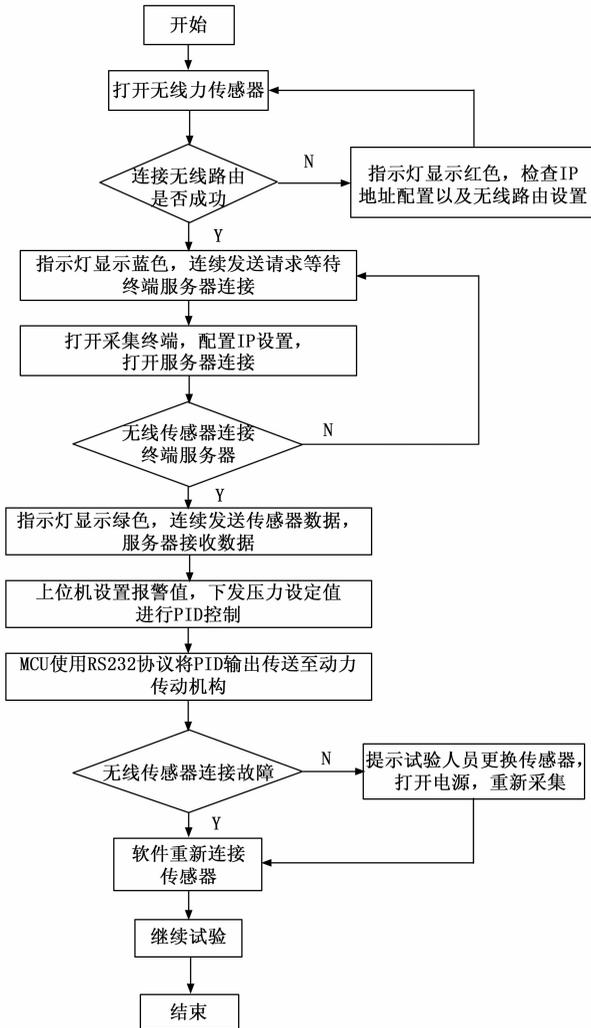


图 2 系统运行流程图

2.1 信号调理电路设计

工业现场长距离传输模拟信号, 如采用电压信号, 传输线的分布电阻会产生电压压降, 导致数据存在不确定的零漂^[7], 为了解决以上问题, 传感器的信号使用 4~20 mA 电流信号, 4 mA 表示零信号, 20 mA 表示信号满刻度。由于 ADC 采集模块的输入范围是 0~5 V, 使用运放搭建如图 3 的信号调理电路, 将电流信号转换为电压信号, 电路的前

端是电流检测放大电路, 使用固定阻值将采集的电流转换为电压信号, 通过差分放大电路, 输出的电压 $U_A = (R_5/R_3) * (I_{\text{检测}} * R_1)$, R_5 和 R_3 取值相等, $U_A = I_{\text{检测}} * R_1$, R_1 的值取 100 Ω , U_A 的范围是 400~2 000 mV; 电压跟随电路产生的电压 U_B , 用来调节基准信号, U_A 和 U_B 作为减法电路的输入, 输出电压为 $(R_{11}/R_9) * (U_A - U_B)$, R_{11} 与 R_9 取值相等, 即值等于 $U_A - U_B$, 当输入的电流 $I_{\text{检测}}$ 为 4 mA 时, 调节高精度电阻 R_8 , 使得 $U_A = U_B = 400$ mV, 减法输出电压为 0 V, 完成基准校准的过程; 当输入电流 $I_{\text{检测}}$ 为 20 mA 时, 减法电路的输出为 $(I_{\text{检测}} * R_1) - U_B = 1 600$ mV, 输出的量程为 5 V, (R_{14}/R_{13}) 为电压放大倍数, 调节 R_{14} 高精度电阻, 使得 U_C 的电压为 5 V, 完成了输入 4~20 mA 电流至 0~5 000 mV 的线性电压输出。

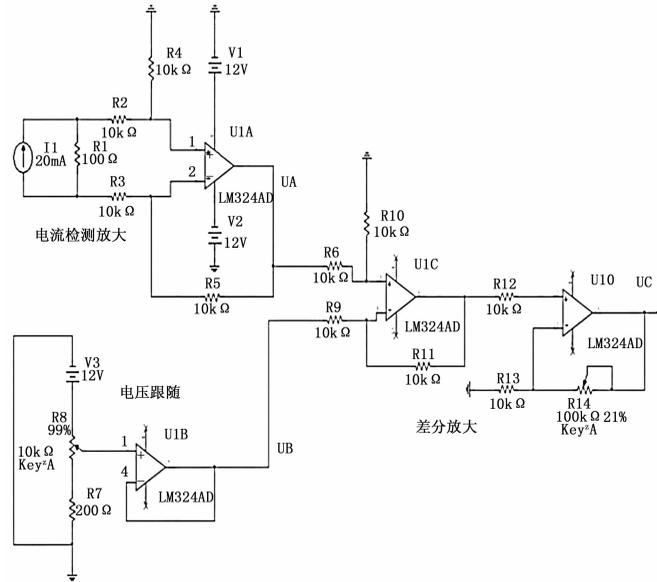


图 3 信号调理电路

2.2 ESP8266 无线模块电路设计

ESP8266 是由乐鑫智能平台提供的一款高性能无线模组, 支持 softAP 模式 (无线接入点), station 模式 (无线终端) 以及 softAP+station 共存模式, 程序使用 softAP 模式进行多点组网^[8-9]。STM32 微处理器的管脚 PB10 与 ESP8266 的管脚 UTXD 相连, PB11 与 URXD 相连, 使用 USART 串口与 ESP8266 的数据进行收发。

参考乐鑫提供的 AT 指令集, STM32 使用串口发送指令字符串对 ESP8266 做通讯, 如表 2 所示。

表 2 AT 指令集

命令	描述
AT+CWMODE	选择 WIFI 应用模式
AT+CWJAP	加入 AP
AT+CWQAP	退出与 AP 的连接
AT+CIPSTART	建立 TCP 连接
AT+CIPSEND	发送数据
AT+CIPMODE	设置模块传输模式

志位为 1 则代表 TCP 连接关闭, 不为 1 则代表 TCP 保持连接。

主循环中执行连接状态判断、串口数据透传和 PID 计算控制 PWM 占空比输出的任务。判断 TCP 连接标志位, 如果保持连接, 将 ADC 采集的电压数据使用串口向 ESP8266 发送数据, ESP8266 将数据通过网络连接发送至电脑服务器, 如果连接断开, 退出透传模式, 获取连接状态, 确认失联后重新连接, 直至下次连接成功。无线传感器在发送数据的过程中, 动态地接收上位机下发 PID 压力设置值和 PID 参数设置值, PID 的初始设置值开机从 FLASH 内存中读取上次的保存值, 使用定时器设置 PID 的计算周期, 在 PID 计算周期中, 使用 ADC 读取的压力实时值与设置值相减得到偏差, 对偏差进行增量式的比例, 积分, 微分算法, 计算出 PID 控制值, 该值为 PID 控制输出引脚 PWM 高电平输出占空比的值, 输出引脚连接至 MOS-光耦隔离开关电路, 动态控制动力传动机构的运行速度, 将压力值控制至设置值。固件程序开发流程如图 5 所示。

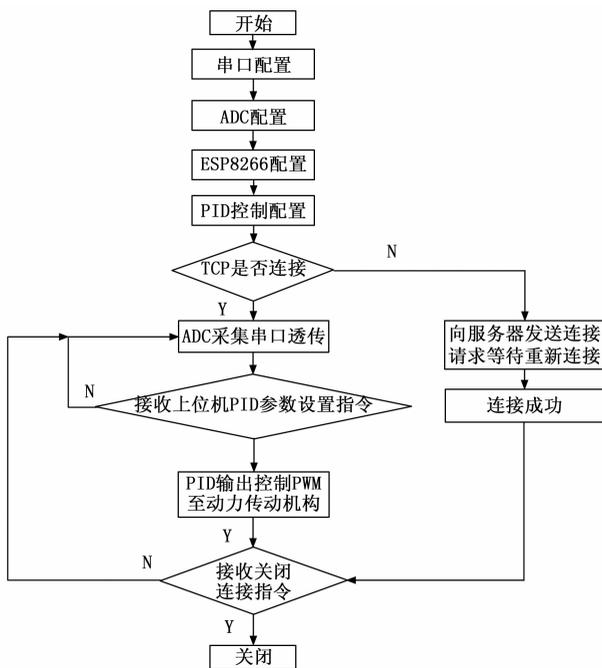


图 5 固件程序开发流程图

3.2 增量式积分分离 PID 算法

工业控制过程中, 被控对象实时采集的数据与设定值之间产生偏差, 对偏差使用比例、积分和微分做控制的算法, 形成 PID 算法, PID 控制具有原理简单, 鲁棒性强和实用面广等优点, 被广泛应用^[13], 图 6 为 PID 算法的示意图。

其中 S_v 为用户设定值, 即控制的目标值, P_v 是控制对象的当前压力值, 偏差 E 是 S_v 减去 P_v 的值, 代表了负载的当前值与目标值的误差程度。历史偏差值是系统投入运行以来, 控制算法每隔一段时间读取传感器的压力值 P_v , 用 S_v 减去 P_v 形成多个偏差 E 的数组; 当前偏差是当前时

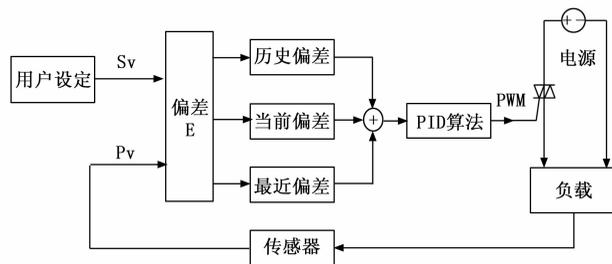


图 6 PID 算法示意图

刻的偏差 E 值, 以单个浮点数字表示; 最近偏差是当前时刻最近的两到 3 个控制周期获取的偏差数组。3 个偏差作为 PID 控制算法的输入, 输出已调的脉冲宽度 PWM 波形至光耦-MOS 开关电路, 动态控制传动机构的开关占空比, 最终将压力控制在设定值。

系统的读取周期使用 T_d 来表示, 每个控制周期读取控制对象的压力值用 X 来表示, $0 \sim N$ 个周期读取的压力数组可以表示为 $[X_{1T_d}, X_{2T_d}, X_{3T_d} \dots X_{(N-2)T_d}, X_{(N-1)T_d}, X_{NT_d}]$ 。第 n 个读取周期的误差 $E_{nT_d} = S_v - X_{nT_d}$, 历史偏差数组序列可以用 $[E_{1T_d}, E_{2T_d}, E_{3T_d} \dots E_{(n-2)T_d}, E_{(n-1)T_d}, E_{nT_d}]$ 来表示。 E_{nT_d} 的值大于 0, 则说明控制未达标; E_{nT_d} 的值等于 0, 则说明控制已达标; E_{nT_d} 的值小于 0, 则说明控制已超标; 引入常量 K_p , 输出 $P_{OUT} = K_p \times E_{nT_d}$, 输出指的是 PWM 周期的占空比, PWM 的周期用 T_k 表示, 即系统的控制周期, 电压处于高电平的时间为 $T_H = T_k \times P_{OUT}$, 在该时间段内, MOS 开关处于导通状态, 传动电路处于全功率输出状态。比例控制是按照偏差大小来修正输出值, 当用户设定值和当前压力值相等的时候, E_{nT_d} 偏差为 0 时, $P_{OUT} = 0$, 比例控制不起作用, 为了避免以上情况的发生, 在比例控制输出值上叠加一个常数 P_{OUT0} , $P_{OUT} = K_p \times E_{nT_d} + P_{OUT0}$, 即使 E_{nT_d} 偏差为 0, P_{OUT} 不等于 0, 比例控制仍然作用, 单独使用比例控制加入常数 P_{OUT0} 。

历史偏差数组序列和用 S_N 来表示, $S_N = \sum_{i=0}^N E_{iT_d}$, E_{iT_d} 的值可正可负, 序列和 S_N 为有限值。第 n 个读取周期的偏差数组序列和用 S_n 来表示, S_n 的值大于 0, 即在过去的 n 个读取周期中, 大多数的输出控制信号没有将当前对象的压力值控制到设定值; S_n 的值小于 0, 即在过去的 n 个读取周期中, 大多数当前对象的压力值超过设定值; S_n 的值等于 0, 意味着在过去的 n 个读取周期中, 总体控制达标。对之前的值做求和, 即积分算法, 使用 S_n 的值产生一个积分控制输出信号 I_{OUT} , 引入常量 K_p , $I_{OUT} = K_p \times S_n$, 在积分控制输出值上叠加一个常数 I_{OUT0} , $I_{OUT} = K_p \times S_n + I_{OUT0}$, 可有效的抵消外界环境因素的干扰。

在第 n 个读取周期 T_d 时, 最近的两个偏差相减, 得到偏差之差 $D_{nT_d} = E_{nT_d} - E_{(n-1)T_d}$, 偏差的变化率 $VD_{nT_d} = (E_{nT_d} - E_{(n-1)T_d}) / T_d$, 当 VD_{nT_d} 的值大于 0, 即偏差值变大, 偏离控制目标程度大; 当 VD_{nT_d} 的值小于 0, 即偏差值变小, 偏离控制目标程度小; VD_{nT_d} 的值用来预测未来的偏

差走势, 速率变大, 适当的控制输出信号减弱趋势, 速率变小, 适当的控制输出信号增强趋势, 产生微分算法控制的效果。使用 VD_{nTd} 的值产生一个微分控制输出信号 D_{OUT} , 引入常量 K_p , $D_{OUT} = K_p \times VD_{nTd}$, 微分控制是不可以独立行动的, 当 VD_{nTd} 等于 0 的时候, 变化率是直线, 没有变大的趋势也没有变小的趋势, 偏差始终存在, 微分算法只考察偏差是否变化, 并不考察偏差是否存在, 控制系统偏差即使很大, 偏差变化为 0, 单独使用微分控制, 不会产生输出。为了在微分控制变化率没有改变的情况下, 系统依然受控, 在微分控制输出值上叠加一个常数 D_{OUT0} , $D_{OUT} = K_p \times VD_{nTd} + D_{OUT0}$, 增强控制的稳定性。

根据以上结论, 推导 PID 算法的数学模型:

$$PID_{OUT} = P_{OUT} + I_{OUT} + D_{OUT} \quad (1)$$

其中: $P_{OUT} = K_p \times E_{nTd} + P_{OUT0}$, $I_{OUT} = K_p \times S_n + I_{OUT0}$, $D_{OUT} = K_p \times VD_{nTd} + D_{OUT0}$, 带入表达式 (1) 中, $P_{OUT0} + I_{OUT0} + D_{OUT0}$ 是常数, 使用常数 OUT_0 来表示, 化简得到式 (2) 如下:

$$PID_{OUT} = K_p \times (E_{nTd} + S_n + VD_{nTd}) + OUT_0 \quad (2)$$

其中: S_n 可以用如式 (3) 表示:

$$S_n = \frac{1}{T_i} \times \sum_{i=0}^n E_{nTd} \times T_d \quad (3)$$

其中: T_i 是积分时间, 在程序中, 采样周期、读取周期及控制周期都是 T_d , PWM 的输出周期使用 T_{pwm} 来表示, T_d 并不是固定的, 根据实际情况变化。假设先去除 D_{OUT} 的影响, 只有 P_{OUT} 和 I_{OUT} 的作用, 构成 PI 控制器, 当系统偏差 $E_{nTd} < 0$ 的时候, 且 $S_n < 0$, P_{OUT} 和 I_{OUT} 认为在过去的这段时间内, 系统未达标, P_{OUT} 和 I_{OUT} 选择共同作用, 共同作用的时间称之为积分时间 T_i , T_i 和 S_n 的值成反比, T_i 越大, 积分输出 I_{OUT} 越小。

偏差之差 VD_{nTd} 的值用式 (4) 表示如下:

$$VD_{nTd} = T_{vd} \times ((E_{nTd} - E_{(n-1)Td}) / T_d) \quad (4)$$

其中: T_{vd} 为微分时间常数, T_{vd} 与微分项作用 D_{OUT} 成正比, T_{vd} 设置的越大, D_{OUT} 越大。为了方便分析, 去除 I_{OUT} 。当传感器的回读值 P_v 并没有达到控制系统的设定值 S_v , 且偏差在向增大的趋势发展, 比例输出 P_{OUT} 和微分输出 D_{OUT} 同时做增大调整, P_{OUT} 和 D_{OUT} 共同运行的时间称为微分时间常数 T_{vd} 。在表达式 (4) 中, 计算周期 T_d 出现在分母上, T_d 的值取的越大, 偏差变化率越小, 不能充分体现控制系统变差的变化程度, 在程序设计微分算法输出 D_{OUT} 时, T_d 的值不宜取的过大^[14]。

将表达式 (4) 中的 S_n 和表达式 (3) 中的 VD_{nTd} 带入表达式 (2), 得到 PID_{OUT} 输出, 如表达式 (5) 所示:

$$PID_{OUT} = K_p \times E_n T_d + \frac{K_p}{T_i} \times \sum_{i=0}^n E_{nTd} \times T_d + K_p \times T_{vd} \times ((E_{nTd} - E_{(n-1)Td}) / T_d) + OUT_0 \quad (5)$$

在表达式 (5) 中, 计算得到的 PID_{OUT} 的值用于表示 PWM 的脉冲宽度, 这种调控方式成为位置式 PID, 优点是可直接输出控制值; 缺点是需要累加大量的历史偏差值,

导致 MCU 的运算工作量过大, 不适合多通道的控制^[15]。

为了避免位置式 PID 过程中计算工作量大等问题, 引入增量式 PID 算法。增量式 PID 是计算控制量的增加值, 即当前输出相对于上个计算周期输出的变化值 ΔPID_{OUT} 。第 $n-1$ 个计算周期的 $PID_{OUT(n-1)}$ 的表达式 (6) 如下所示:

$$PID_{OUT(n-1)} = K_p \times E_{(n-1)Td} + \frac{K_p}{T_i} \times \sum_{i=0}^{n-1} E_{(n-1)Td} \times T_d + K_p \times T_{vd} \times ((E_{(n-1)Td} - E_{(n-2)Td}) / T_d) + OUT_0 \quad (6)$$

第 n 个计算周期的 ΔPID_{OUT} 用 PID_{OUTn} 减去 $PID_{OUT(n-1)}$, 化简后如表达式 (7) 所示:

$$PID_{OUT(n-1)} = K_p \times (E_{nTd} - E_{(n-1)Td} + \frac{K_p}{T_i} \times E_{nTd} \times T_d + K_p \times T_{vd} \times ((E_{nTd} - 2E_{(n-1)Td} + E_{(n-2)Td}) / T_d)) \quad (7)$$

在表达式 (7) 中, E_{nTd} 表示第 n 次计算周期的偏差, 只需要 E_{nTd} 、 $E_{(n-1)Td}$ 表示第 $n-1$ 次计算周期的偏差, $E_{(n-2)Td}$ 表示第 $n-2$ 次计算周期的偏差, 增量式 PID 只需要计算 E_{nTd} 、 $E_{(n-1)Td}$ 和 $E_{(n-2)Td}$ 的偏差值, 相比于位置式 PID 中的大量过去偏差值之和, 减少了单片机的资源占用量和计算工作量。位置式与增量式 PID 控制的区别如下:

1) 输出不同: 位置 PID 控制的输出需要使用整个整定过程中累加的误差值, 数据量大; 增量 PID 的输出仅与当前周期和前两周期的误差有关, 计算量少, 不过多占用系统内存;

2) 积分部件: 采用增量 PID 控制, 为控制量增量输出, 未采用积分算法, 适用对象需带有积分部分, 位置 PID 适用的执行对象, 不需要积分部件。

3) 记忆功能: 增量 PID 是控制量增量输出控制, 当系统出现故障, 则影响较小, 执行器具备记忆功能, 不会严重影响系统的工作, 而位置输出不具备记忆功能, 直接使用近几个周期的误差对系统输出, 对系统有较大的影响^[16]。

为了解决系统的超调问题, 引入积分分离的控制方式。系统控制开始时, P_v 和 S_v 之间存在偏差 E , 且 $E > 0$, 在 P_v 和 S_v 第一次相等之前, 所有的偏差之和都是正数相加, 导致偏差之和比较大, 偏差之和与积分输出 I_{OUT} 成正比, 系统此刻已经达标, P_{OUT} 不再起作用, I_{OUT} 的输出效果加强, 在 PI 控制中会产生过冲超调的现象, P_v 会超出 S_v , 偏差 $E < 0$, 偏差之和由于加入负数, 逐渐减小, I_{OUT} 的输出效果也逐渐减小, 最终 P_v 趋近于 S_v 。因此在控制系统中, 为了避免过冲超调现象的出现, 在 P_v 和 S_v 第一次相等之前, 将积分时间 T_i 设置的非常大, $1/T_i$ 无限接近 0, P_{OUT} 与 $1/T_i$ 成正比, 积分输出也接近于 0, 忽略积分项的作用, PID 算法不断的对偏差进行判断, 当 P_v 接近 S_v 的 80% 时, 修改 T_i 的值, 加入积分作用, 避免积分项过早的投入运行, 有效的减小偏差之和, 减小过冲超调的效果, 这种方式称为积分分离, 即将积分算法在早期分离出去, 不用历史数据来对当前对象做调整。在 P_v 和 S_v 第一次相等之后, 积分算法体现出优势, 控制系统一旦达标, 比列项不产生

作用, 积分项使用历史偏差产生控制信号。

总结如下, 比例算法只考虑当前是否有偏差, 与输入偏差成比例关系, 系统一旦存在偏差, 比例控制产生作用以减小偏差, 单独使用比例控制, 系统始终存在稳态误差, 称为有差系统。积分算法是对系统之前的偏差进行累加积分, 时间越长, 积分作用越大, 减小稳态误差, 直至偏差为零。积分控制用于减小系统的稳态误差, 提高系统控制准确性, 使用积分控制, 导致相位滞后, 系统易发生超调。因此, 引入积分分离的方式, 有效减小系统误差积分累加值, 从而控制的超调值。使用微分算法, 输出与偏差的变化率成正比关系, 提前控制输出信号的变化趋势, 减小超调量。PID 控制结合了比例、积分、微分 3 种控制算法的优点, 适当选取 K_p 、 K_i 和 K_d 参数的数值, 产生良好的控制质量, 具有较大的应用范围^[17]。

3.3 上位机程序整体设计

为了防止数据传递的丢失及线程耦合产生竞争关系, 重新设计了程序框架, 程序使用 QMH 框架 (QueueMessageHandle 线程队列处理), 该框架是基于生产者/消费者队列机制搭建^[18], 如图 7 所示, 在寄存器中开辟独立栈进行事件或数据的队列入栈操作, 执行网络配置、网络连接、数据处理存储解析等一系列事件, 实现了事件线程之间的解耦合, 确保了程序的稳定性。

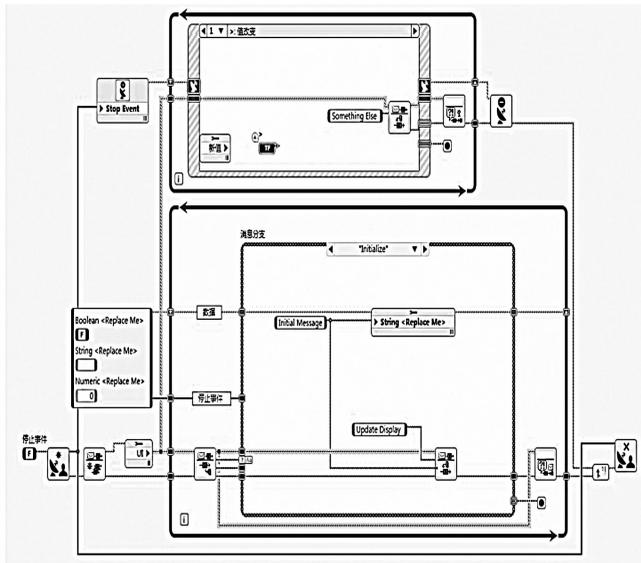


图 7 软件设计框架

NI 公司基于 NI-PSP 协议, 提供了网络变量库 (Network Variable Library), 旨在解决测控系统分布式数据同步的问题, 用户可以在多个分布式系统或程序间做数据同步, 且不影响程序的运行性能^[19-20];

SVE 引擎 (SharedVariableEngine) 用于共享变量的网络发布及部署, SVE 通过如图 8 所示链路所示部署发布该值, 在 B/S 系统中, SVE 客户端起到共享变量服务器的作用。

在航天器机构展开任务压力传感器参数测控系统中,

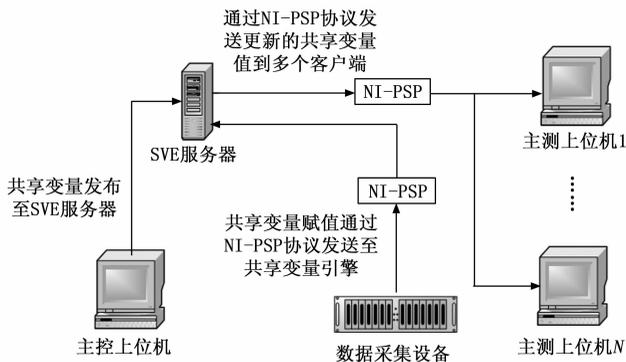


图 8 NI-PSP 服务器协议共享变量发布接收链路

主控上位机负责试验的主要测控任务, 大数据的存储读取显示处理会占有线程的运行周期, 从而延长试验的 PID 控制周期, 导致系统运行缓慢, 压力控制超调, 稳定误差过大, 将试验数据通过共享变量的形式使用 NI-PSP 协议发送至 SVE 服务器, 多台主测电脑同步从 SVE 服务器读取试验共享变量数据做处理, 不影响整个试验的测控速度, 实现一主多测的试验模式, 实现 PID 的快速控制, 限制压力超调, 保证控制精度。

无线数据采集软件采用模块化编程, 根据测控需求将软件分为试验登录、试验配置、试验过程、试验数据分析及试验维护等模块, 每个模块又分为若干个小的功能模块, 这样设计既方便调试和修改, 又可灵活配置使用于不同应用场景。

试验登录包括用户权限管理及界面选择功能; 试验配置包括试验信息配置、数据存储路径配置、报警信息存储路径配置、服务器网络配置、采集通道零位校准及 PID 参数配置; 试验过程包括: 加载配置表、遍历巡检无线传感器、数据采集存储数据库、动态拟合加速度曲线、动态修改通道设置、动态修改 PID 控制参数、故障反馈重连机制; 试验数据分析包括: 数据库读取、历史数据拼接和自动生成试验报表; 试验维护包括: 用户数据库维护、试验数据库维护、设备启动维护及试验表单维护等, 具体的试验功能划分如图 9 所示。

软件的前面板界面如图 10 所示。

4 试验结果与分析

系统设计开发完成后, 选取典型的压力阶梯控制工况进行试验, 使用多个无线压力传感器连接传动机构和路由器, 开启终端服务器。编写工况列表, 程序根据设置时间, 自动切换被控对象的压力为 10 kg、20 kg、30 kg, 做压力阶梯控制试验, 分别使用常规 PID 算法和改进的积分分离 PID 算法试验结果如图 11 所示。

常规的 PID 算法, 未能及时有效地消除积分项带来的超调偏差, 有损坏仪器设备的风险。10 kg 压力控制工况时, 最大实测值为 12.68 kg; 20 kg 压力控制工况时, 最大实测值为 22.71 k; 30 kg 压力控制工况时, 最大实测值为

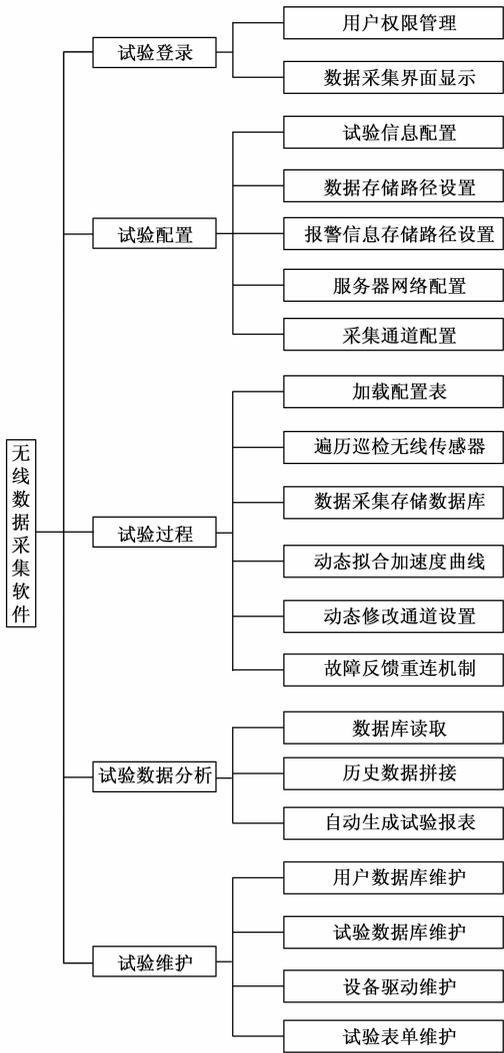


图 9 软件功能模块

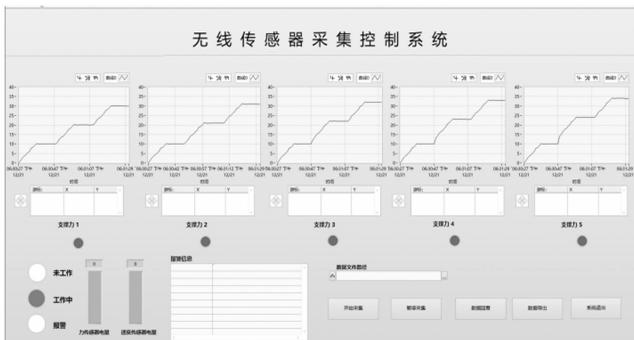


图 10 软件界面设计

30.66 kg, 超差率较大, 不符合设计要求。采用改进的积分分离增量式 PID 算法, 在 PID 控制开始阶段, 设定了积分分离的百分比, 在控制刚开始的阶段, 积分算法不参与控制, 当控制值达到了积分分离的设置值后, 积分算法参与控制, 有效的控制历史误差的积分和, 减小超调偏差。不

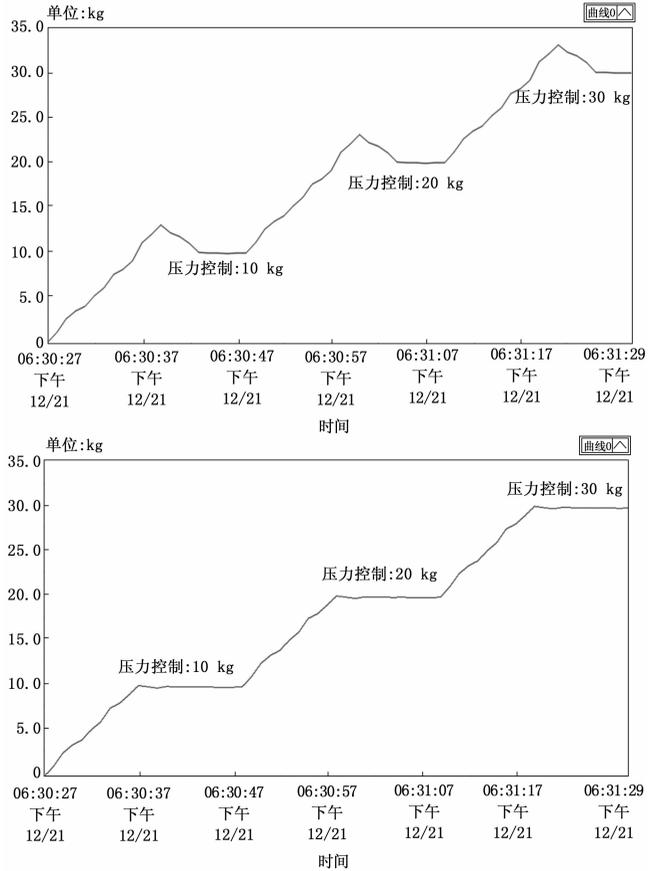


图 11 阶梯压力采集控制试验结果

同的积分分离百分比, 对于不同的控制系统, 产生的控制效果不同, 实测值如表 3 所示。

表 3 实测值对比表

积分分离百分比/%	控制压力/kg	最大实测值/kg
80	10	11.23
90	10	10.46
100	10	10.68
80	20	21.33
90	20	20.41
100	20	20.71
80	30	31.27
90	30	30.42
100	30	30.62

使用 PSP 服务器协议, 有效的较少了 PID 控制周期, 提高了控制精度, 同时减小了超调误差, 选用 90% 的积分分离百分比, 控制效果最佳, 试验的超调误差不超过 0.5 kg, 动态稳定误差不超过 ±0.4 kg, 符合试验大纲的设计要求。

5 结束语

基于 ESP8266 和 LabVIEW 虚拟仪器的嵌入式分布式无线压力传感器参数测控系统, 使用 STM32 微处理器、ESP8266

(下转第 29 页)