

同濟大學

TONGJI UNIVERSITY

毕业设计（论文）

课题名称 风电运维用多通道数据采集系统

副标题

学院 机械与能源工程学院

专业 机械设计制造及其自动化

学生姓名 梁宇翹

学号 1851143

指导教师 朱玉田



日期 2023 年 3 月 19 日

装

订

线

摘要

为应对全球气候变化，中国向世界郑重承诺，力争 2030 年二氧化碳排放量达到峰值，努力争取 2060 年实现碳中和。风能发电作为目前可再生能源中重要的一部分，具有储量丰富、低排放低污染等特点。风机叶片作为风力发电机捕获风能的关键构件，其对风电机组的正常运转具有无可替代的作用。叶片在运行时一旦发生损伤，不仅会影响风电机组正常运转、降低发电效率，甚至可能导致机组整体崩塌，发生严重安全事故。为此对风机叶片运行的相关数据采集和健康监测极为重要。风机叶片长度较长，对叶片的实时监测需要连接大量传感器并进行长距离数据传输。使用常见的数据采集系统可能出现采集数据不稳定、传输速率过慢、线缆过于复杂、成本过高等问题，因此针对风力发电运维开发多通道数据采集系统具有重大意义。

针对风力发电运维问题，搭建了风电运维多通道数据采集系统，提出了基于 GD32 微处理器与 W5500 网络通信模块的总体方案，采用网络通信的方式回传采集数据，使用 LabView 接收并进行数据处理，实现了多通道数据高速采集、网络通信、数据存储等功能。

分析目前已有数据采集系统方案，对比其优点与不足，确定风电运维数据系统的设计要求，明确总体方案，包括采集系统硬件、软件开发方案；完成了对采集系统中数据采集卡的硬件设计，包括主控芯片、通讯模块选型以及电路整体方案设计；针对系统采用的嵌入式系统进行程序开发，完成了数据采样、数据发送功能；完成系统上位机软件开发，实现数据接收、波形显示与数据存储功能；最后，搭建了采集系统实物，进行数据采集测试并做了进一步优化。

实物测试结果显示，所搭建的风电运维数据采集系统十分有效，能很好完成对风机叶片运行状况的实时监测，满足风电厂运行维护要求。

关键词：风能发电，风机叶片，数据采集系统，网络通信

Multi-channel data acquisition system for wind power operation and maintenance

ABSTRACT

In response to global climate change, China has made a solemn commitment to the world striving to peak CO₂ emissions by 2030 and working towards carbon neutrality by 2060. As the most important part of renewable energy, Electricity from Wind is characterized by abundant reserves, low emissions and low pollution. As the key component of wind turbine to capture wind energy, wind turbine blade has an irreplaceable role in the smooth operation of wind turbine. Once the blade in operation occur damage, will not only affect the normal operation of wind turbines, reduce power generation efficiency, and may even lead to the collapse of the whole unit, serious safety accidents. Therefore, the wind turbine blade operation of the relevant data collection and health monitoring is extremely of great importance. Wind turbine blade length is fairly long, for the blade of the real-time monitoring need to connect a large number of sensors and long distance data transmission. The use of common data acquisition systems may lead to unstable data collection, slow transmission rate, too complex cabling, high cost and other problems, so the development of multi-channel data acquisition system for wind power operation and maintenance is of great significance.

A multi-channel data acquisition system for wind power operation and maintenance is built,. The general scheme based on GD32 microprocessor and W5500 network communication module is proposed. Network is used to transmit the collected data back, and LabView-based programs are applied to receive and process the data, which fulfills the functions of high-speed multi-channel data acquisition, network communication and data storage.

Firstly, the existing data acquisition system schemes are analyzed. Comparing their advantages and shortcomings, the design requirements of the wind power operation and maintenance data system are determined, and the overall scheme is clarified, including the hardware and software development plans of the acquisition system; then, the hardware design of the data acquisition card in the acquisition system is completed, including the selection of the main control chip, communication module and the overall circuit scheme design; secondly, the embedded system adopted for the system is developed, and the data sampling and data sending functions are completed; the software development of the system upper computer is accomplished in order to fulfill the data receiving, waveform display and data storage functions; finally, the acquisition system was constructed physically, tested and further optimized.

The actual test results show that the constructed wind power operation and maintenance data collection system is very effective and can well complete the real-time monitoring of the wind turbine blade operation condition and meet the operation and maintenance requirements of wind power plants.

Key words: Wind power generation, wind turbine blades, data acquisition systems, network

communication

装

订

线

目 录

1	引 言	6
1.1	风力发电及风机叶片运维检测概况	6
1.2	多通道信息采集技术概况	6
1.3	国内外方案比较	7
1.4	本文所做的工作	8
2	风电运维用多通道数据采集系统方案设计	9
2.1	总体方案设计	9
2.2	系统硬件方案	9
2.2.1	微处理器方案	9
2.2.2	数据采集方案	10
2.2.3	通讯方案	11
2.2.4	供电方案	12
2.3	系统软件方案	13
2.3	系统整体方案	13
3	系统硬件设计	14
3.1	主控芯片选型	14
3.2	通讯模块选型	14
3.3	供电方案	15
3.4	地址设置方案	15
3.5	电路整体方案	15
4	嵌入式软件设计	17
4.1	开发方案	17
4.2	系统架构设计	19
4.2.1	系统任务分析	19
4.2.2	系统时序处理	19
4.3	采样频率调整、数据采样与 DMA 存储功能	22
4.4	网络通讯功能	23
4.5	数据打包功能	25
5	上位机软件设计	26
5.1	上位机要求	26
5.2	系统上位机软件主函数	26
5.3	数据读取线程	27
5.3.1	数据读取线程总循环	27
5.3.2	字符串解包	29
5.4	文件保存线程	30
5.4.1	文件保存线程总循环	30
5.4.2	波形生成与数据存储	30
6	系统测试与优化	34
7	总结与展望	38
	参考文献	39
	谢 辞	41

1 引言

1.1 风力发电及风机叶片运维检测概况

全球可再生能源发电装机容量中风电占有压倒性优势，在被利用的可再生能源中风能占了一半以上，而风力发电也是可再生能源应用技术中最为领先的^{[1][2]}。风能发电作为一种清洁能源，相较于火电等具有建设周期短、储量丰富、低排放低污染的特点，近年来得到世界各国重视并快速发展。截至 2019 年年底，我国已建成 4000 多个风电场，拥有 12 万台风电机组，全国风电累计装机 2.1 亿 kW^[3]，风力发电机组的单机容量已从最初的几十千瓦发展为今天的几百千瓦甚至兆瓦级。风电场也由初期的数百千瓦装机容量发展为数万千瓦甚至数十万千瓦装机容量的大型风电场。随着风电场装机容量的逐渐增大，以及在电力网架中的比例不断升高，对大型风电场的科学运行、维护管理逐步成为一个新的课题。

风机叶片是风力发电机的核心部件，其成本占到风力发电机总成本的 15~20%^[4]，负责承压并将风能转化为机械能传递给发电机。叶片在工作时工况较为复杂，可能受到雨水侵蚀、强风、雷击等损害^[5]，故在运行过程中，叶片的健康监测尤为重要。Amenabar 等^[6]介绍了用于风机叶片无损检测的各项技术。张宇等^[7]根据风力机叶片振动频谱图的各个频段范围，对叶片是否发生故障进行诊断；王艳青^[8]通过采集风机叶片应变时的声发射数据实现对叶片疲劳损伤的检测。要实现风机叶片实时监测，需要采集声信号、振动信号多维度信息，此外叶片尺寸大，尤其在长度方向，目前超过 100 米的叶片已大量装机，需要大量传感器，故设计用于风电叶片运维监测的多通道采集系统，成为叶片运维急需解决的问题。

1.2 多通道信息采集技术概况

一个完整的数据采集系统由原始信号、信号调理设备、数据采集设备以及上位机数据采集软件构成。大部分情况下现实中的原始物理信号并非直接可测得的电信号，需要传感器将物理信号转换为电流或电压信号供数据采集设备处理。数据采集设备的作用在于将传感器传入的模拟电信号转换为数字信号存储并转送至计算机处理。针对风电叶片运维监测情景，对数据采集系统有以下要求：

1 模拟信号采集功能

2 多通道模拟量输入

采集系统需要连接大量传感器，输入通道数需尽可能多。

3 数据高速远程传输功能

多通道传感器采集的大量数据需要及时上传至主机，考虑风机叶片长度，数据必须方便读取、远距离传输不衰减。

4 成本低

考虑单个叶片所需大量信号采集模块，必须保证信息采集系统在保证性能功能前提下成本适中

5 安装便携

信息采集系统需要安装于风机叶片上，过于复杂的通信线缆不仅难以安装、耗费大量材料，

同时也无法保证可靠性，线路应尽可能简单。

1.3 国内外方案比较

数据采集系统起源于 20 世纪 50 年代，美国首先研究了用于军事上的测试系统。随着集成电路设计制造技术的提高，出现了高性能、高可靠性的单片数据采集系统。H. E. Gad 等人设计的针对太阳能设备的温度采集系统^[9]使用 PIC18F46K20 微处理器作为上位机芯片，搭配 DS1302 实时时钟芯片通过串行外设接口将当前时钟和控制信号发给下位机，实时检测下位机状态并控制下位机读取传感器数据并按日期格式存储。下位机使用 ATmega2560 微处理器，采用板上集成的 ADC（Analog-to-Digital）模块将传感器读取得到的模拟信号值转换为数字量，并通过 SPI 将数据存储至 SD 卡中。该方案为独立系统，不需要通过 USB 总线额外连接 PC 也能进行数据采集。上位机配有独立的备用电源，选用的低功耗芯片可以保证在外部供电意外断开时能够正常记录数据，下位机拥有丰富引脚资源，支持多达 16 路模拟量输入，可连接 16 路传感器。国内张月^[10]提出采用 FPGA（Field Programmable Gate Array，现场可编程逻辑门阵列）外接 TI 公司的 8 通道 24 位高精度模数转换模块 ADS1278，针对水声信号进行高精度采样，将数据通过 SPI 交由 STM32 芯片存储至 SD 卡中，得到稳定可靠、高精度、低功耗的采集存储系统方案。该方案为 FPGA 预留了频率选择拨码开关，可通过硬件方式控制 FPGA 的分频系数，将晶振时钟分频到不同频率上作为 A/D 转换模块的时钟信号 CLK，可在 5, 10, 15, 20 kHz 4 种采样频率间切换。但上述提到的数据采集方案均为单一独立系统，在风机叶片运维监测此类特殊应用场景下需要大量数据采集模块，且需要远程读取数据功能，SD 卡数据存储方式不适用该场景。国外市场上拥有很多数据采集系统产品，比如美国 NI 公司生产的数据采集系统采样频率高达 1.25MHz，但采样通道往往偏少，价格较为昂贵。美国安捷伦公司生产的 U2300A 系列 USB 模块化多功能数据采集设备采用模块化设计，所有模块采用通用的高速 USB2.0 接口，支持 64 路模拟输入通道，通道采样率 200kHz 到 1MHz 不等，价格也高达上万元。李冰洋^[11]提出的麦克风阵列声源定位系统方案同样采用 USB2.0 传输协议进行数据传输，传输速率可以达到 480Mbps。但 USB 最长传输距离 5 米，超过这个距离就会有信号衰减，不适用于目标工况。国内北京阿尔泰科技发展有限公司生产的 PCI8814 动态信号采集卡是一款为测试音频和振动信号而设计的高精度数据采集卡，提供 8 路同步模拟输入通道，24bit 分辨率，单通道采样速率高达 102.4kSPS。每通道集成独立的 IEPE 激励源，可实现加速度传感器及麦克风等相关的信号调理。该公司的 NET2991 多功能数据采集卡提供 16 路单端输入或差分输入，16 位 ADC 采样精度。数据经过 FPGA 处理后通过板载以太网 PHY 芯片，采用 UDP 协议与主机进行信息交互，可实现大量数据远距离传输。

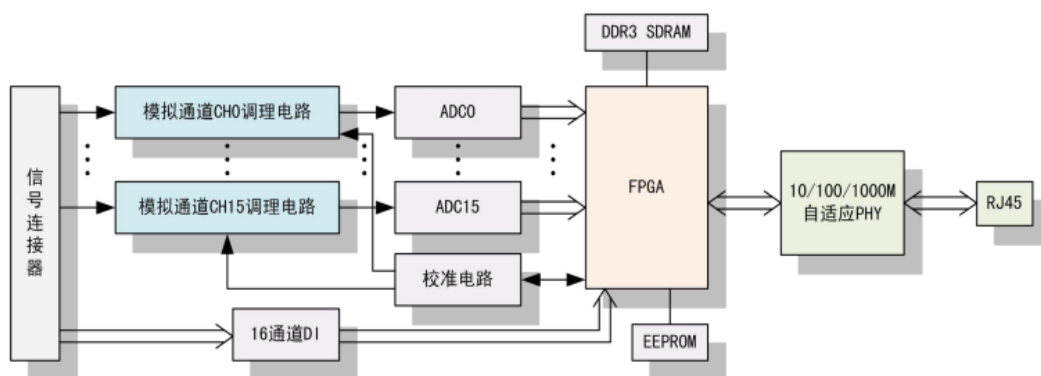


图 1-1 NET2991 系统框图

工业级数据采集卡多采用 FPGA 作为主控芯片，开发难度较大，同时成本较为昂贵。目前市面上暂无针对风电叶片运维监测情景、成本低廉、带数据高速远程传输功能、安装便携可靠的数据采集系统，故本文针对这一特定场景设计一款数据采集系统。

1.4 本文所做的工作

设计用于风电叶片运维监测的多通道采集系统，采用单片机实现单点至数点的信号采集，通过总线通信，将数据传至上位机（工控机），再由上位机对全部的数据进行处理与分析，以便得到有用的健康监测数据，实现对风电叶片运维的多通道采集监测。

本次毕业设计主要围绕以下几个部分展开：

第一章：引言部分，介绍目前数据采集系统相关研究现状，对比各方案优缺点，确定风电运维系统的设计要求。

第二章：总体方案设计，根据所要求明确系统硬件、软件开发方案。

第三章：硬件方案设计，明确主控芯片、通讯模块选型以及电路整体方案设计。

第四章：嵌入式软件设计，进行数据采样、网络通信等功能开发。

第五章：上位机软件设计，实现数据接收、波形显示与数据存储功能。

第六章：系统测试与优化，完成数据采集系统实际测试，并进行优化

第七章：总结与展望

2 风电运维用多通道数据采集系统方案设计

2.1 总体方案设计

数据采集系统一般有以下组成部分：传感器（负责诸如震动、声音、光线等信号的采集，将实际物理信号转为电信号）、数据采集卡（负责接收传感器回传的电信号并打包发送）、上位机（通过总线或网络的方式接收数据采集卡回传的数据包并进行分析处理）。

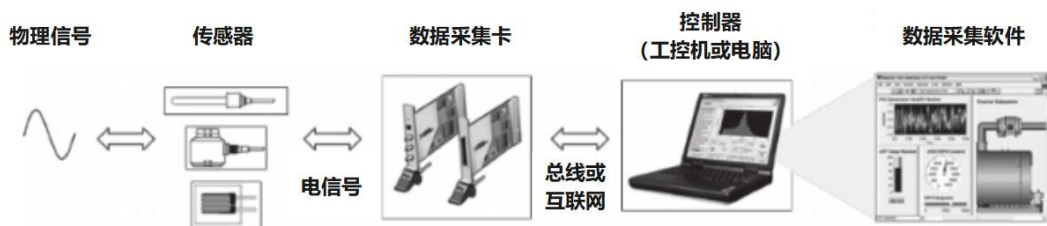


图 2-1 数据采集系统的基本组成^[12]

目前多路数据采集系统的设计有 2 种方法，一种是基于下位机单片机和上位机 PC 的系统架构方式进行设计，优点是下位机硬件成本较低，缺点是上位机软件开发较为复杂，下位机嵌入式使用的是传统的面向过程的 C 语言，针对不同需求需要单独开发，周期较长^[13]；另一种方案是基于 NI 的数据采集卡以及上位机 LabVIEW 图形化编程语言的方式进行开发^[14]，优点是开发难度小、周期短、可以针对用户需求对系统做出快速更改，缺点是 NI 公司的数据采集卡较贵，需要同时采集大量数据、安装大量传感器时成本过于高昂^[15]。考虑实际工况，风机叶片尺寸大，需要大量布设传感器对单一叶片进行检测，成本压力较大，故应采取下位单片机加上位机 PC 的方案开发系统。

2.2 系统硬件方案

2.2.1 微处理器方案

自研数据采集系统常使用微处理器对数据进行处理，内集成 FLASH、SRAM、E2PROM、定时器/计数器、UART、ADC 转换、SPI 等功能，几乎包括数据采集和控制中所需要的所有功能模块，完全可以胜任一般数据采集系统微控制器的设计要求，大致分为三种技术路线，一种是早期使用 STC51 系列单片机进行开发^[16]，优点是成本低廉、资料全，缺点是作为入门级单片机的资源都不及 stm32，例如主频率、ROM、RAM、IO 端口和外围资源等，且其羸弱的性能无法使用 RTOS（实时操作系统），开发方式较为传统；一种是使用由意法半导体公司开发的、针对高性能、低成本、低功耗嵌入式应用设计的单片机 STM32 芯片^{[17][17]}，其拥有更丰富的外设资源，性能更为强大，可以使用实时操作系统，开发较为方便快捷，但近期受国际局势影响，价格波动较大；一种是使用 FPGA 开发，FPGA 外设较少，但可以同时运行多条指令，主频远高于 STM32，适合高速数字处理，相对应价格也比 STM32 更为昂贵，同时功耗更高。

表格 2-1 微处理器方案对比

微处理器方案	优势	劣势
STC51 单片机	成本低廉	性能羸弱、外设资源匮乏、无法使用实时操作系统、开发较为繁琐
STM32 单片机	成本适中、性能适中、可以运行实时操作系统	受国际形势影响，价格波动较大
FPGA	功能强大、可拓展性强	成本较为昂贵、开发困难

考虑实际工况应采用 STM32 单片机。但近期 STM32 芯片价格波动剧烈^[18]，需考虑价格更为稳定、廉价的国产替代方案。国内芯片公司兆易创新 GigaDevice 研发推出的 GD32 单片机与 STM32 定位类似，均为低成本高性能的 32 位微处理器，不同点在于 GD32 的主频最大可达 108M，而 STM32 主频最大 72M，意味着 GD32 代码运行速度会更快，性能相较于 STM32 更为强劲，同时内核电压仅为 1.2V，STM32 内核电压位 1.8V，GD32 的运行功耗更低。

2.2.2 数据采集方案

对于数据采集系统而言其核心功能是数据转换，这是一个将模拟信号转换为数字格式的过程，反之亦然。数据转换需对信号进行量化后再进行编码。量化意味着用一组离散值表示模拟信号的连续值，而编码意味着用编码的形式表示这些离散值。常见的数据采集方案中都会使用专用的芯片实现 ADC 转换器（Analog Digital Convector，模拟信号到数字量转换器）。ADC 转换器有以下主要技术参数：

1. 分辨率

分辨率表示输出数字量变化一个相邻数码所需要输入模拟电压的变化量。通常定义为满刻度电压与 2^n 的比值，其中 n 为 ADC 的位数。例如具有 12 位分辨率的 ADC 能够分辨出满刻度的 $1/1212$ （0.0244%），位数越高代表分辨率越高。

2. 量程

量程指能转换的输入电压范围。

3. 转换时间

转换时间是 ADC 完成一次转换所需要的时间，即从启动信号开始到转换结束并得到稳定的数字输出量所需要的时间，通常为微秒级。采样信号频率越高，所需的 ADC 采样速率也应越高。

除了 AD（模拟量转数字量）转换功能外，ADC 要实现数据采样，还需要一个系统定时控制器产生所需的时钟和定时信号，以准确、精确地控制采集的时间。

项目选用的 GD32 型单片机上集成了 12 位逐次逼近式模数转换器模块（ADC），可以采样来自于 16 个外部通道和 2 个内部通道上的模拟信号。这 18 个 ADC 采样通道都支持多种运行模式，采样转换后，转换结果可以按照最低有效位对齐或最高有效位对齐的方式保存在相应的

数据寄存器中。片上的硬件过采样机制可以通过减少来自 MCU 的相关计算负担来提高性能。综合考虑数据使用 GD32 单片机板载 ADC 采样模块进行采集。

2.2.3 通讯方案

通讯分并行通讯与串行通讯。并行通讯是指所有数据位同时进行传输的通讯方式，通讯所需线缆数与数据位数相等，其特点是传输速度快、效率高，但成本昂贵，抗干扰能力差，通常只适合 30m 距离内、对实时性有较高要求的数据传输中。串行通讯是指通讯双方按位进行，遵循时序的通信方式，最少仅需两条线缆就能完成信息交换的功能，特点是线路少、布线简单、可根据实际工况选择所需通讯协议^[19]。考虑实际工况，通讯使用线缆应尽可能少，数据应采用串行总线的方式传输。下面列出常见的通讯总线与物理接口标准。

表格 2-2 常见通讯总线对比

通讯总线名称	形式特点	传输速率
UART	全双工异步通信，分为 RX、TX、GND 三根信号线，协议结构较为复杂，抗干扰能力差，通讯距离极短，常用于计算机与串行设备的串口通讯	较慢
I2C	半双工同步通信，分为 SDA、SCLK 两根信号线，协议复杂但所需线缆较少，抗干扰能力较弱，较少用于远距离通信	较快，标准模式下可达到 100Kbit/s，高速模式最高可达 3.4Mbit/s
SPI	全双工同步通信，分为 SDO、SDI、SCLK、SS 四根信号线，硬件较为复杂但协议简单，传输数据位电压电平受距离干扰大，可靠传输距离仅 1~3m	快，部分性能较强的器件通讯速率可达 50Mbit/s

表格 2-3 常见物理接口对比

物理接口名称	形式特点	传输速率
RS232	采用单端传输，逻辑电平范围大，抗干扰能力较强，但受限于共模干扰传输距离较短（15m），只允许一对一通信	较慢，标准传输速率最大仅为 19.2Kbit/s
CAN	传输可靠、距离远（最远可达 10km），实时性高，结构简单，对线缆没有特殊要求，系统成本低，但高速率下通讯距离短（小于 100m）	快，传输速率可达 1Mbit/s
RS485	采用差分信号传输，抗干扰能力强，通讯距离长（可达 1500m），允许多点组网，但在高速率（1Mbit/s）下传输距离不超过 100m。	较快，最高传输速率可达 10Mbit/s

Ethernet	协议较为复杂,成本较高,但兼容性好,成本低、易组网、数据传输距离长。	极快,经典以太网最高传输速率可达3Mbit/s,交换式以太网最高支持10000Mbit/s（称为万兆以太网）
----------	------------------------------------	--

风机叶片运维检测时将使用麦克风对声信号进行采样,声信号频率在 5~10kHz,采样位数按 12bit 计算,采样频率不得低于 10kHz,需同时采样 16 通道数据。则总线传输速率要大于 $10\text{kHz} \times 12\text{bit} \times 16 \approx 2\text{Mbps/s}$,同时数据必须满足长距离传输不衰减的要求,综合考虑通讯总线选择 SPI 协议,使用以太网进行数据传输。

以太网是一种计算机局域网技术,它使用双绞线或光纤等物理介质来连接不同的计算机设备,实现数据的传输和共享。国际互联网标准化组织提出 OSI 参考模型 (Open System Interconnection Reference Model, 开放式系统互联模型)以解决各硬件在相同层次上的信息交流,基于该模型建立的以 TCP/IP 协议为核心的互联网网络体系将以太网分为四层,自上而下分别为:应用层、传输层、网络互连层、网络接口层^[21]。越下面的层,越靠近硬件;越上面的层越接近用户。

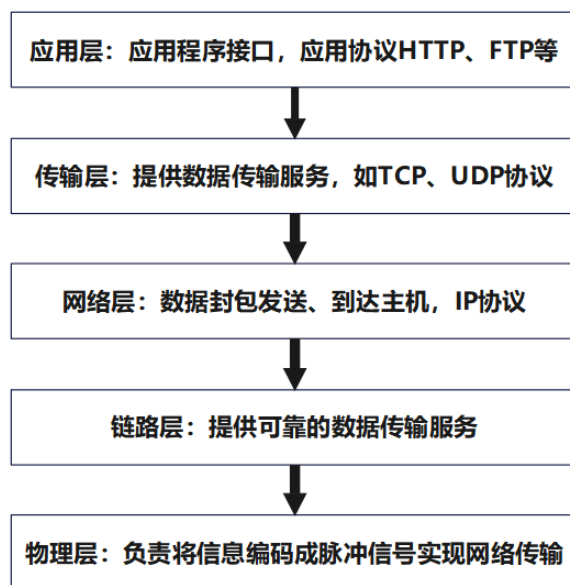


图 2-2 以太网五层模型

搭建以太网实现数据通讯的物理层需要计算机设备（服务器、工作站）、网络连接设备（网卡、集线器、交换机等）以及网络传输介质三部分。系统需连接多数据采集卡,故网络物理层需交换机拓展主机网口,数据采集卡上需拥有网络端口以连接线缆。

2.2.4 供电方案

数据采集系统需要多块数据采集板共同工作,同时还需外接大量传感器,这些设备需要外接电源,且分布式安装在风机叶片上,距离主机较远,不仅加大整体网络组件成本、布线复杂,同

时加大了网络故障的可能性。针对这一问题，IEEE 推出 PoE 标准（Power over Ethernet，以太网供电）。

以太网供电是指在现有网络基础设施中引入电源供电，允许以太网在传输数据时提供 DC 电源，为小型网络设备（如电话、数据终端等）供电，从而简化布线，降低网络基础设施的建设成本^[21]。PoE 系统包括 PSE（Power Source Equipment，供电设备）和 PD（Powered Device，用电设备）两部分。PSE 有两种方式为设备供电^[22]：中跨式（Midspan），在交换机外部增加拓展设备，接入交换机输出的以太网线，通过双绞线将网线中未使用的备用线给用电设备供电；端跨式（Endspan），以太网交换机中集成供电控制电路，电源从以太网端口隔离变压器的中芯抽头或备用线为终端用电设备供电。系统选择带 PoE 功能的交换机为采集系统供电。

2.3 系统软件方案

用数据采集系统上位机程序使用 Qt 或 LabView 语言开发。Qt 是一个用于桌面、嵌入式和移动设备应用程序的 C++ 开发框架，支持包括 Linux、OS X、Windows、Android、iOS 等平台。LabView 是 NI 公司开发的应用于 Windows 平台的图形化编程环境，是专门针对数据采集和控制系统设计而开发的虚拟仪器开发平台软件。LabVIEW 提供的信号处理功能，可以对数据采集板所采集风机叶片相关信号进行分析处理。综合考虑选择 LabView 进行上位机程序开发。

2.3 系统整体方案

确定系统整体方案如下：

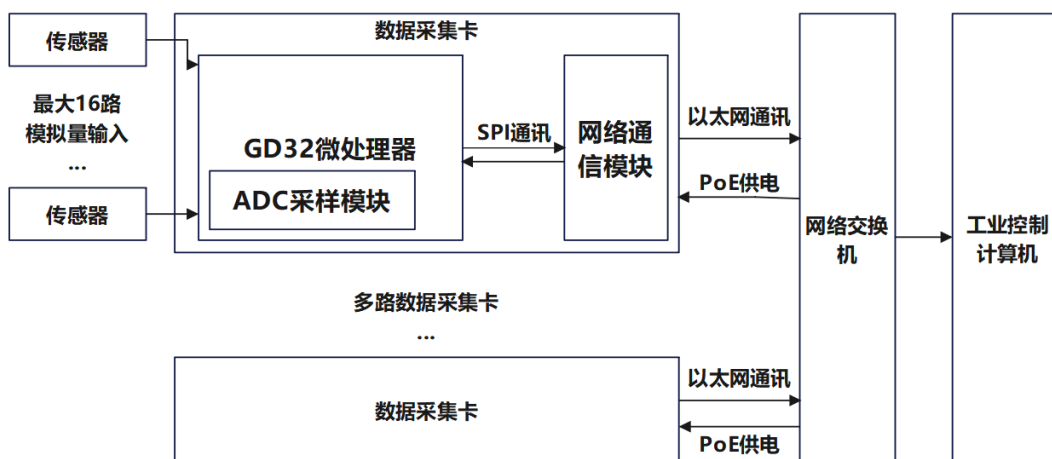


图 2-3 系统整体方案

3 系统硬件设计

数据采集板使用嵌入式系统执行读取传感器数据、网络通信等任务^[23]。

3.1 主控芯片选型

系统中主控芯片的主要功能是调用 ADC 模块快速转换外接传感器信号，并通过 SPI 总线将采集数据发送至网络通信模块。因此要求选用的主控芯片允许至少 16 通道模拟信号输入、主频率尽可能高以实现快速信息处理、ADC 采样尽可能快、拥有必要的中断资源、定时器、SPI 引脚等，还要求微控制器成本尽可能低、拥有足够容量的内部存储器。

系统选用 GD32F303VET6 作为主控芯片对风机叶片数据采集系统进行设计，设置主频为 120Mhz，使用其 2 路 12 位分辨率 ADC、DMA、USART 串口通信与 SPI 通信等功能。

3.2 通讯模块选型

嵌入式系统使用以太网需解决以下问题：TCP/IP 协议的实现，数据封装、编码与发送，数据冲突检测和重发功能。嵌入式以太网大致分为以下几类技术方案^[24]：

1. 采用高性能微处理器+网络接口控制器+精简 TCP/IP 协议栈

大幅简化标准 TCP/IP 协议栈以降低其对系统资源的要求，实现在低成本、内存空间紧张的嵌入式系统上以太网通信的部署^[25]。使用专用的网络接口控制器芯片（Network Interface Controller，简称 NIC）协同解决 OSI 参考模型中数据链路层的功能^[26]。该方案的难点在于 TCP/IP 协议栈的精简，要求开发者对单片机底层与计算机网络有较为深入了解，开发难度较大。

2. 使用硬件协议栈芯片

硬件协议栈芯片是一个将 TCP/IP 协议与链路层协议以硬件方式实现的、具有独立上网功能的专用芯片，通过输入接口与微处理器进行通讯即可实现以太网部署。该方式有效降低了开发难度与开发周期。

表格 3-1 以太网通讯方案对比

TCP/IP 协议	嵌入式 TCP/IP 协议	硬件协议栈芯片
应用层	简单实现或不实现	全部由硬件实现，用户只需调用接口与其通讯即可
传输层	嵌入式处理器根据自身特点进行选择实现	
网络层		
链路层	应用网络接口控制器芯片完成	
物理层	驱动电路	

采用硬件协议栈方式实现以太网在嵌入式系统的部署。韩国 WIZnet（微知纳特）公司开发的 W5500 是一款全硬件 TCP/IP 协议栈以太网接口芯片，用户无需在软件层面对以太网进行配置，可直接调用硬件实现以太网通讯。芯片通过高速 SPI 协议与微处理器通讯，实现微处理器网络通信功能。芯片集成 8 个独立的硬件 Socket，各路通信互不影响；拥有 32KB 片上缓存供 TCP/IP 包

处理；集成以太网 MAC 与以太网 PHY；支持 Linux & RTOS 等嵌入式操作系统。综合考虑，选择 W5500 作为通信模块核心芯片。

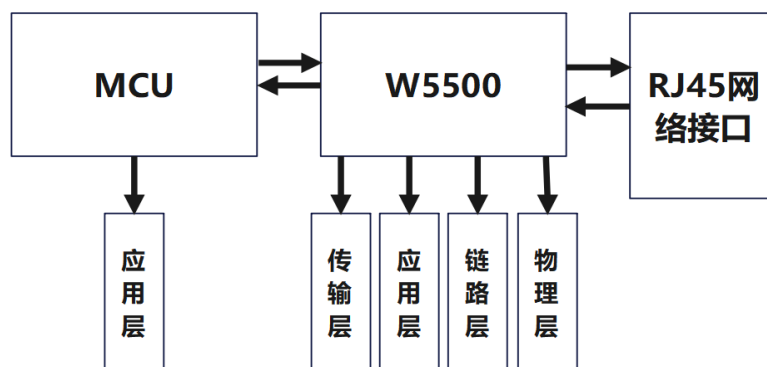


图 3-1 W5500 实现全硬件 TCP/IP 协议栈以太网接入方案

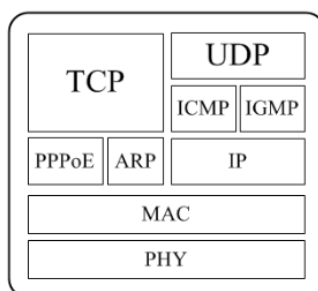


图 3-2 W5500 芯片内核

3.3 供电方案

PoE 供电功能需使用注入 Linear 的 LTC4257、TI 公司的 TPS2383 等 PD 接口控制器进行供电接口设计，较为繁琐。目前市面上已有专用 PoE 分离器，无需用户设计电路即可实现供电与网络通信功能，用户只需提供 DC 接口供电源连接、RJ45 接口供网线连接即可。故供电方案采用外接 PoE 分离器。

PoE 供电电压一般为 12V，系统选用的传感器（如麦克风等）一般工作于 5V，GD32F303VET6 微处理器工作电压为 3.3V，故数据采集板需降压稳压模块。

3.4 地址设置方案

由于系统需多块数据采集板共同工作，每块板子需有不同 IP 地址，防止串联网通信时数据冲突。网络地址的修改可以在软件代码里修改，但修改后需重新烧录程序，不适合大规模使用。故应增设硬件网络地址修改功能，通过改变硬件拨杆、接线等方式即可实现 IP 地址的修改。数据采集板采用跳线帽接地的方式设置编号，上电后采集板会读取引脚状态设置编号。

3.5 电路整体方案

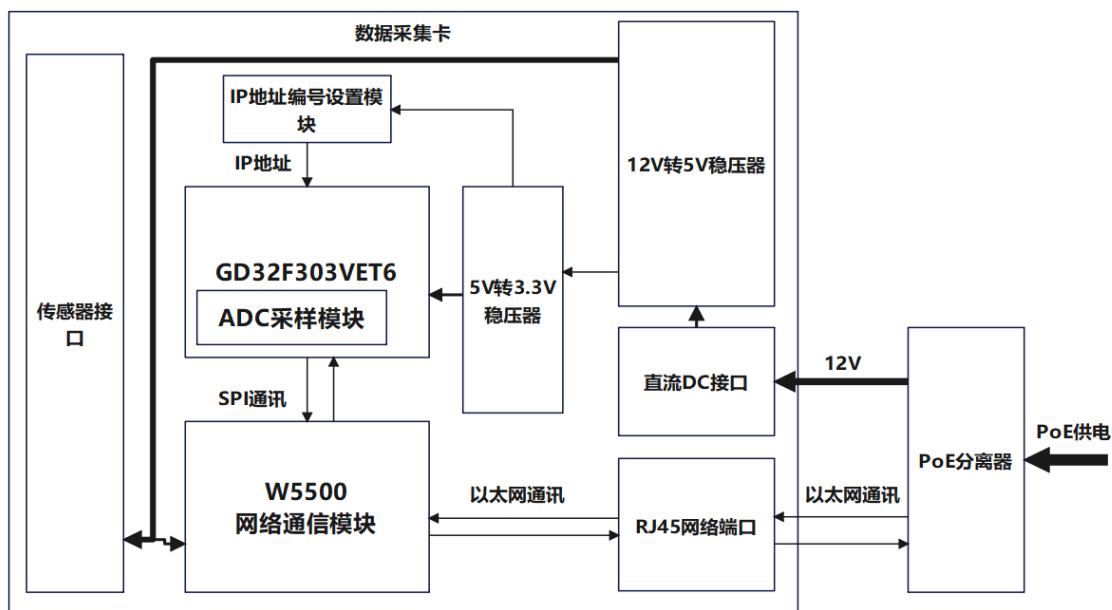


图 3-3 数据采集版电路示意图

装
订
线

4 嵌入式软件设计

嵌入式软件使用 Keil 进行开发，导入兆易公司提供的 GD32F303 芯片功能包即可进行开发。使用 Github 网站作为代码管理工具，记录各版本文件。

4.1 开发方案

GD32 作为对标 STM32 系列单片机的国产化替代方案，其开发方式同样与 STM32 兼容。对于 STM32 系列嵌入式系统开发而言分为以下三种开发模式：

1. 基于寄存器的开发

寄存器开发是直接操作 STM32 的底层寄存器，这种方式最接近硬件，但也最复杂，需要熟悉芯片的数据手册。

2. 基于标准库函数的开发

库函数开发是使用芯片公司提供的标准外设库（Standard Peripherals Library），这种方式是对寄存器的封装，提供了一些宏定义和结构体变量，方便用户配置外设。

3. 基于 HAL 库函数的开发

HAL 库开发是使用 ST 公司推出的硬件抽象层（Hardware Abstraction Layer），这种方式是对库函数的进一步抽象，提供了一些通用的 API 函数接口，方便用户在不同的 STM32 系列产品之间移植。HAL 库还包含了一些第三方的中间件，如 RTOS，USB，TCP/IP 等。



图 4-1 STM32 三种开发模式关系

表格 4-1 STM32 三种开发方式对比

开发方式	程序执行效率	开发难度
寄存器开发	最高。直接进行硬件操作，理论上可以针对寄存器进行代码优化，最灵活。在针对大量数据处理时寄存器开发的优势明显。	最难。用户需对芯片底层硬件由全面的认知，需经常翻阅芯片数据手册，操作寄存器工作十分繁琐，工作量随程序复杂程度成倍增加。同系列不同型号芯片寄存器存在较大差异，代码无法通用，移植困难。

库函数开发	中等。对寄存器操作进行封装，存在部分冗余操作。	适中。开发者容易应用函数完成程序开发，无需关心底层寄存器调度，但仍需对整体硬件系统有一定了解。库函数在同系列芯片通用，移植难度适中，但无法移植至不同系列芯片。
HAL 库开发	最低。HAL 库针对不同系列芯片标准库都做了抽象，代码空间占用较大，执行效率较低，功能封装过于复杂。	最低。HAL 库开发无需对芯片硬件系统有深入了解，调用提供的程序接口即可实现各外设的公共函数功能，轻松实现从不同 STM32 系列产品代码的移植。

ST 公司针对 HAL 库开发推出了图形化代码生成工具 CubeMx，通过图形化界面，可以非常直观的对片上外设，时钟，中断，DMA 等等各种设备的参数进行配置，CubeMX 可以直接生成初始化代码，使得开发人员可以将更多的精力放在核心代码的开发上。

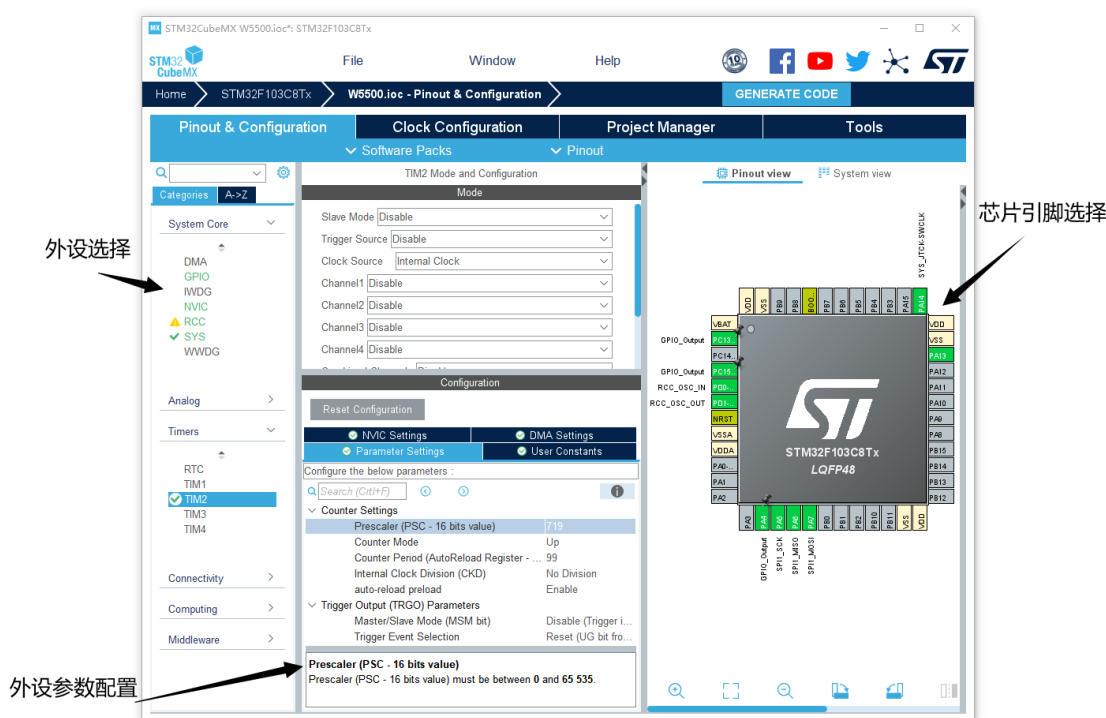


图 4-2 STM32CubeMx 图形化界面

从开发难度考虑，HAL 库+STM32CubeMx 更适合数据采集系统的嵌入式开发。尽管 GD32F30x 系列芯片与 STM32F10x 系列芯片是 PIN TO PIN 兼容的（即封装形式、引脚位置与功能都是一一对应的），芯片内部寄存器、逻辑地址等完全相同，但 GD32 部分寄存器默认值不同，外设模块的设计时序与 STM32 存在差异，尤其是网络通讯等较为复杂功能下需要针对 GD32 做出的修改移植工作量较大且难度较高，经尝试后放弃在 GD32 使用 HAL 库开发的方案，选择可移植性较差、需要对底层硬件有一定了解但开发难度适中、稳定可靠的库函数方案，使用兆易创新提供的库函数进行嵌入式系统开发。

4.2 系统架构设计

4.2.1 系统任务分析

数据采集板嵌入式部分需完成以下功能：

1. 读取设置 IP 地址

要实现该功能，应当配置对应编号设置引脚，将引脚电平值转换为 IP 地址，使用该参数初始化网络通信。

2. 采样频率调整、数据采样与 DMA 存储

ADC 的采样频率需使用定时器控制。定时器中断控制 ADC 模块进行采样，调整定时器中断时间即可调整 ADC 采样频率。

微处理器读取引脚电平，调用 ADC 模块将 引脚电平模拟值转换为对应的数字量编号。由于采样频率较高，频繁调用网络通信函数会占用较多时间，此外需同时采 16 通道数据，数据量较大，需要调用 DMA（Direct Memory Access,直接内存访问）功能以满足大容量数据的高速访问。后文会针对该问题做进一步阐述。

3. 串口通信

串口通信是嵌入式系统开发中必不可少的功能，通过串口通信可以将单片机运行状态、内部变量等少量数据直接回传至上位机，方便系统测试开发。

4. 网络通信

微处理器需进行 SPI 通信等配置，同时需根据网络通讯需求对 W5500 芯片进行相应设置，ADC 数据也需要按一定格式打包发送。

4.2.2 系统时序处理

由于系统采样频率较高，普通单线程工作方式难以满足数据采集与传输的要求。具体而言，数据采集系统对时序要求较为严格，数据打包、传输等工作需占用一定时间，在系统频率较高、多核心处理器或采样频率要求较低时可以忽略这部分用时，但对于性能有限的单片机而言数据打包、网络通信等用时较长，导致 ADC 无法按原定频率进行数据采集，回传信号会有明显丢失的问题。解决这一问题有以下两种方法：

1. 使用前后台系统

前后台系统，即主函数在死循环中对各个子函数不断循环检测调用，一个任务的运行必须等待上一任务运行结束，所有任务都是平级的。主函数大循环被称为后台，所有任务在无限循环的后台中运行或等待运行条件到来。单片机设置了中断机制来处理紧急任务，中断处理被称为前台。前台程序可以中断后台程序的运行，单片机优先执行前台程序，前台程序执行完毕后再返回后台任务断点处继续执行。

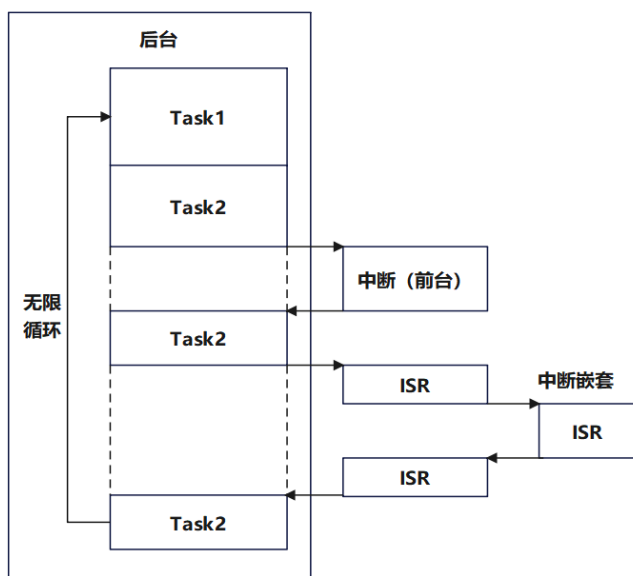


图 4-3 前后台系统

前后台系统可以通过延迟中断的形式对数据进行响应处理。在中断中改变后台任务的状态位，退出中断后，后台循环读取到任务运行条件改变，进行数据处理响应等操作。当任务较为复杂、大量使用中断会占用系统资源，浪费时间。此外，主函数中的任务仍是平级关系，尽管任务状态位已在前台中发生变化，主函数中仍然需要等待上一程序运行结束后才能响应。

结合实际时序分析，后台循环等待中断改变状态位，当 DMA 接收到足够数据时触发中断，改变状态位；程序退出中断后后台循环读取到状态位改变，进行数据处理打包、发送工作并清楚状态位，微处理器在主函数中随时可以被前台打断，执行 ADC 转换工作，保证采集数据的完整性。但如果数据处理打包工作执行时间较长，采样频率较高，DMA 再次触发中断改变状态位时程序无法及时响应，仍可能导致数据丢失问题。

2. 使用实时操作系统 RTOS

RTOS（Real-time operating system，实时操作系统）是一种用于实时计算应用的操作系统，在硬件与应用程序之间提供内核，管理系统的硬件资源，为应用软件提供必要服务^[27]。嵌入式实时操作系统通过实时任务调度器进行 CPU 资源分配，以时间片轮询的方式执行程序，系统可同时执行多个同优先级任务，此外还可以随时被高优先级任务打断，执行完后再继续执行低优先级任务。在众多嵌入式实时操作系统中，FreeRTOS 作为轻量级的开源操作系统，其可移植、可裁剪、调度灵活等特点^[28]适合部署于内存资源紧张的微处理器上。

针对数据采集系统，ADC 数据采集需独立于 FreeRTOS 系统，保证数据采集完整，使用 DMA 双缓冲区的方式保存 ADC 数据，将数据打包、网络通信分为两个独立任务，提高程序运行效率，保证数据的完整性。

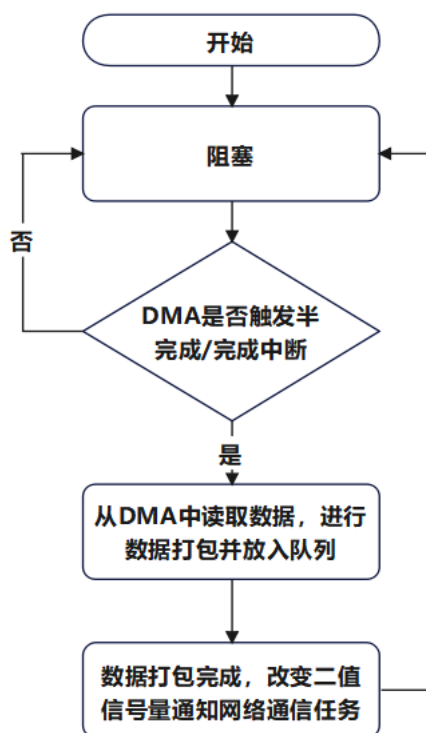


图 4-4 数据打包任务

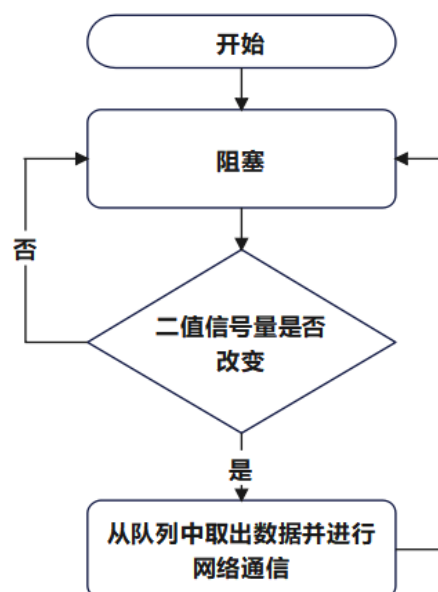


图 4-5 数据发送任务

两个任务执行的条件不同，但优先级相同。系统在执行数据发送任务时出现 DMA 中断，此时数据打包任务也会同步进行，保证数据能及时从 DMA 中读取，完成打包工作，提高系统采样频率上限。

将 FreeRTOS 内核进行必要裁剪，移植至 GD32F303VET6 芯片并进行功能开发后测试得知，理论上实时操作系统的多线程操作确实是能保证实时性，同优先级任务可以同时进行，互不影响。但微处理器本质是单核，通过时间片轮询方式实现的多线程处理本质上仍是单线程。在一般使用下可以忽略 FreeRTOS 切换不同时间片的耗时，但当所需 ADC 采样频率较高，需要使用定时器频繁触发采样时 FreeRTOS 切换时间片的耗时无法忽略。此外，FreeRTOS 依据其设定的系统节拍 configTICK_RATE_HZ 进行工作，默认系统节拍频率为 1kHz，提高该值能加快系统频率，减少时间片轮询切换所需时间，但会显著提高内核负载，影响任务实时性。实际测试 FreeRTOS 无法满足在高达 10kHz 频率触发中断下良好的实时性，回传数据有明显丢失问题，且系统容易卡死，故 FreeRTOS 并不适用于高频数据采集系统开发。

系统最终选择前后台系统进行开发，通过优化代码、开启 DMA 双缓冲区等方式实现最高 10kHz 频率的 ADC 采样与数据完整发送。程序整体流程图如下：

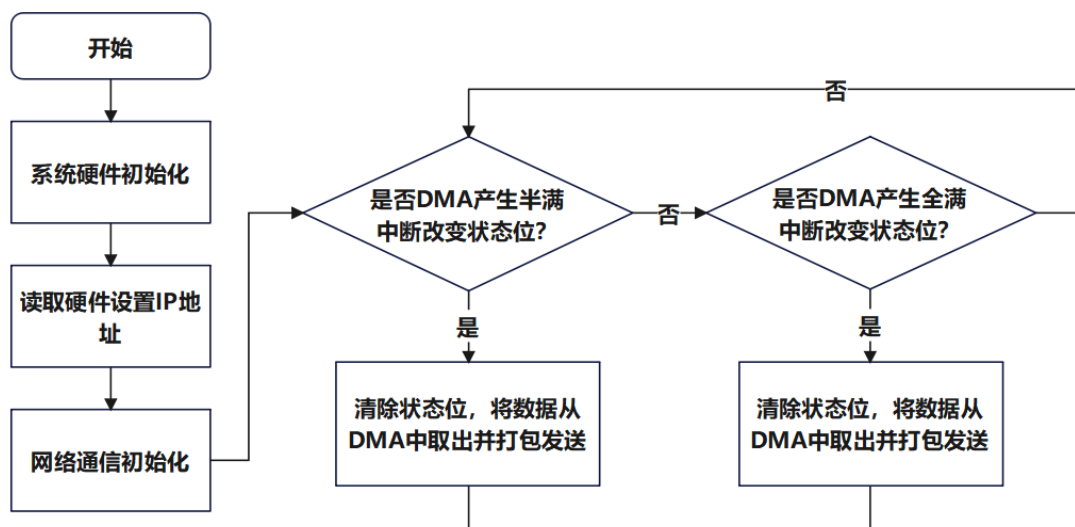


图 4-6 程序整体流程图

4.3 采样频率调整、数据采样与 DMA 存储功能

ADC 拥有注入通道和规则通道两种，注入通道可以中断规则通道的采样，但没有 DMA 请求，需要使用轮询或中断模式来保存转换数据，且最多仅能处理四个通道数据，故选择规则通道进行采样。

为实现 ADC 采样频率调整，需设置定时器作为外部信号输入源进行控制。设置定时器为 PWM 输出模式，当定时器计数至目标值时内部输出高电平，持续目标时间后再拉低，重置计数器，定时器重新开始计数。GD32 系列芯片支持高达 120MHz 系统时钟 ($f_{sys} = 120\text{ MHz}$)，采用定时器 1 作为 ADC 触发源，设置其时钟分频值 (clockdivision) 为一分频 ($div = 1$)，预分频值 (prescaler) 为 119 ($psc = 119$)，计数周期 (period) 为 99 ($arr = 99$)，则其定时器触发周期 T_{tim} 为

$$T_{tim} = \frac{(arr + 1)(psc + 1)}{f_{sys}} = \frac{(99 + 1)(119 + 1)}{120} = 100\text{ us} \quad (4.1)$$

即 ADC 的触发频率 $f_{adc} = \frac{1}{T_{tim}} = 10\text{ kHz}$ 。

ADC 设置为扫描转换模式且失能连续转换模式，接收到定时器给出的 PWM 触发信号后对 16 通道进行扫描、读取电平值，完成扫描后等待下一次定时器触发信号。GD32 拥有多 ADC 模块，可以同时工作以提高采集效率。选择常规并行模式进行采样，ADC0 与 ADC1 交替工作。



图 4-7 ADC 扫描采样示意

DMA（Direct Memory Access,直接内存访问）是一种在外设和存储器之前或存储器与存储器之间进行高速数据传输的技术。开启后，DMA 能有效减轻 CPU 负担，就系统而言 ADC 数据无需 CPU 处理即可直接输入至存储器中，需要处理时 CPU 再从存储器中读取。

为解决数据打包、网络通信用时较长导致 DMA 中未被打包的数据被覆盖的问题，可以使用双缓冲区模式（又称乒乓缓冲机制），即 DMA 对一个缓冲区操作时，其他程序可以对另一个缓冲区操作，数据将在双缓冲区中循环存储。系统通过软件实现双缓冲区机制，开启 DMA 半满与全满中断，ADC 数据传入 DMA 后半部分/前半部分不影响程序对 DMA 中前半部分/后半部分的数据读取与处理。

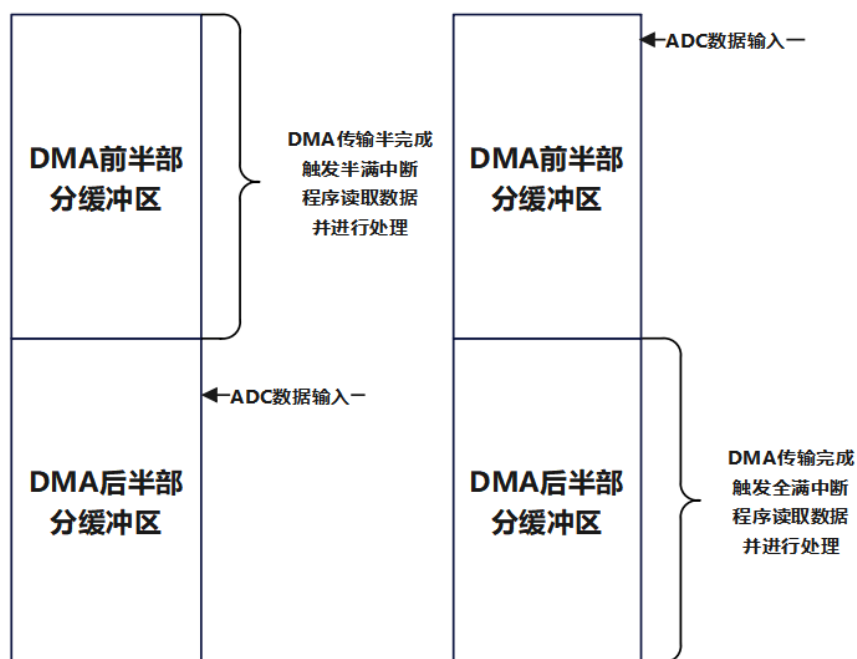


图 4-8 DMA 乒乓缓冲机制

4.4 网络通讯功能

微处理器通过 SPI 协议与 W5500 芯片进行通信，使用前需进行初始化设置。W5500 芯片提

供了网络通信所需的初始化函数，用户修改参数、进行移植即可调用，完成对 W5500 芯片相关寄存器的配置。其中，程序已读取用户设置的 IP 地址。

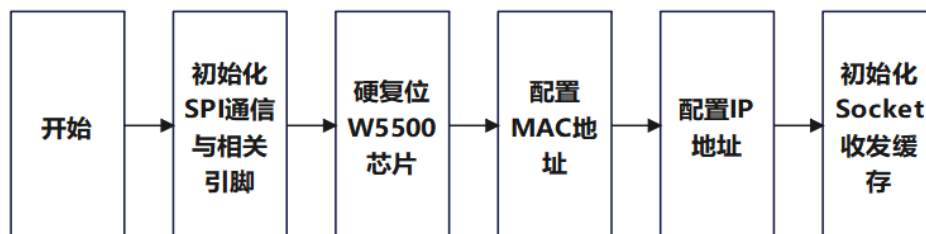


图 4-9 网络通信初始化流程

设定的IP地址为: 192.168.1.123

```

使用定义的IP信息配置W5500
W5500 IP地址   : 192.168.1.123
W5500 子网掩码 : 255.255.255.0
W5500 网关     : 192.168.1.1
W5500可以和电脑的UDP端口通讯
W5500的本地端口为: 3000
远端端口为: 3000
W5500可以通过UDP回传信息至PC端!
    
```

图 4-10 网络通信初始化后回传信息

系统初步使用 UDP（User Datagram Protocol）协议进行网络通讯。UDP 协议是一种无连接的传输层协议，应用程序间无需建立连接即可发送封装的 IP 数据包。其特点在于无需维护主从机的连接状态，服务器可同时向多从机传输相同信息；UDP 信息包标题很短，相较于 TCP 而言额外开销小；吞吐量不受拥挤控制算法的调节，只受传输带宽、主从机性能等限制。相较于 TCP，UDP 拥有更好的实时性与工作效率、更小的网络开销。考虑微处理器与 W5500 芯片性能限制，系统数据采集频率较高，需频繁触发网络通信进行数据传输，选择无需握手、网络开销更小的 UDP 协议进行网络通讯。

Socket 是网络通信中的重要概念，将 TCP/UDP 层复杂的网络通信操作抽象为简单的接口供应用层调用，以实现进程在网络间的通信。Socket 是应用层和传输层中的抽象层，其在网络通信模型中的位置如下：

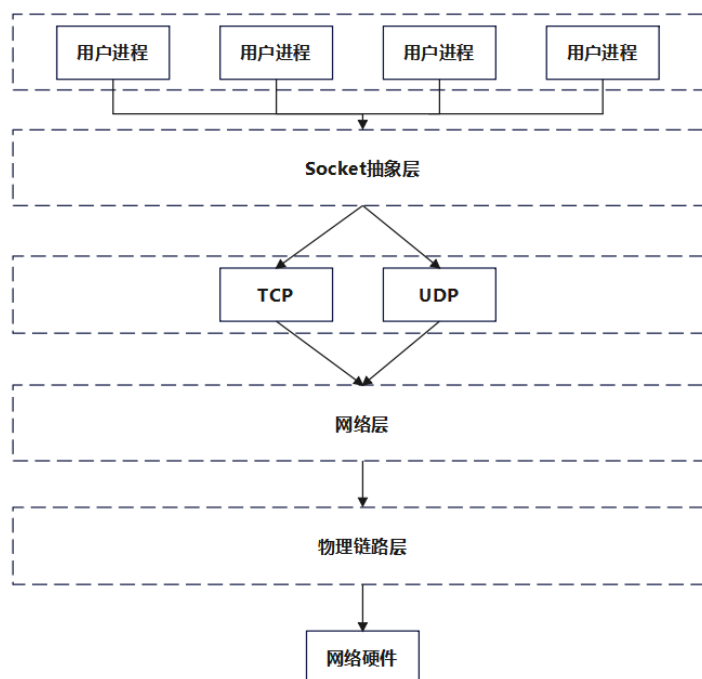


图 4-11 网络通信模型

网络通信初始化后，微处理器可以通过 SPI 通讯读取其 Socket n 配置寄存器 Sn_CR 状态得知其对应 Socket 当前运行状态，根据当前状态选择打开 Socket、调用指定 Socket 端口进行 UDP 通讯。

4.5 数据打包功能

为保证精度，ADC 采集的数据以 32 位的形式存储，且为小端存储（即第一个字节是低位，最后一个字节是高位）。微处理器与 W5500 芯片采用的 SPI 通讯单次仅能发送 8 位数据，且上位机接收数据时为大端存储，在发送数据时需要进行转换。

存储的数据:
0x11223344

地址	0x00	0x01	0x02	0x03
数据	0x11	0x22	0x33	0x44

小端存储

地址	0x00	0x01	0x02	0x03
数据	0x44	0x33	0x22	0x11

图 4-12 大小端存储示意图

为减少频繁网络通信带来的处理器卡顿问题，DMA 缓存区大于单次 ADC 采集数据大小，采集数据达一定大小后启动网络通信，将数据包一次性发回上位机。LabView 上位机 UDP 模块最大单次接收数据量为 548 比特，设置数据采集板单次回传数据包为 512 比特。

5 上位机软件设计

系统使用 LabView 进行上位机软件开发。

5.1 上位机要求

数据采集卡采集数据打包后通过网络通信发回，LabView 需实现 TCP/UDP 通讯功能，监听网络端口接收数据采集卡回传数据，实时显示并进行存储。由于 LabView 中自带的仪器 IO 模块仅支持 NI 公司以及其他少部分公司的专用数据采集卡，使用自制数据采集卡需自行完成通讯协议编写等工作，实现对数据包的解包、转换。

5.2 系统上位机软件主函数

上位机软件主界面如下：

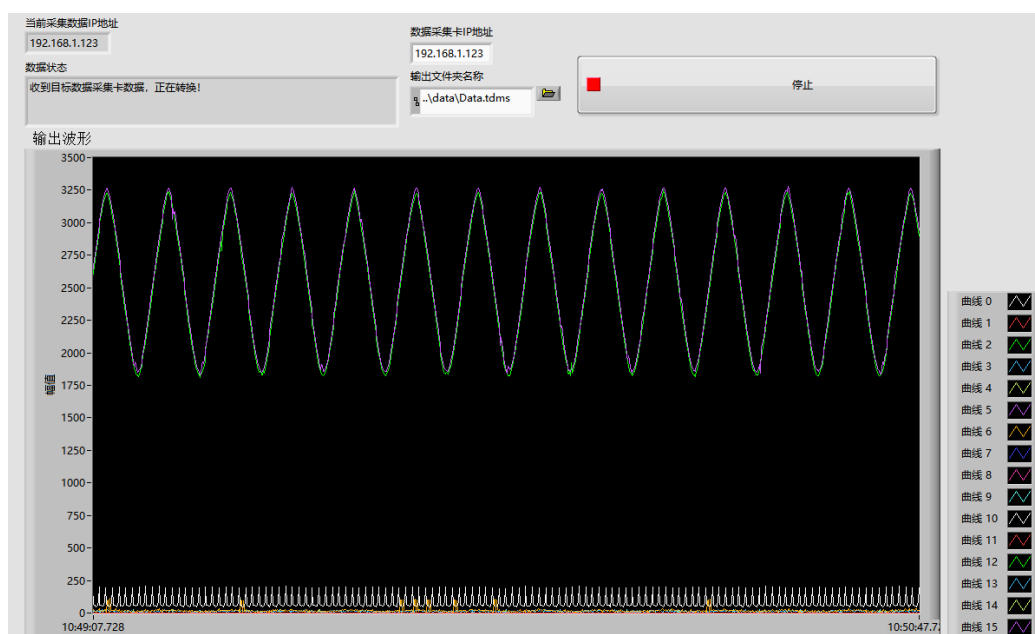


图 5-1 上位机软件界面

系统实现了以下功能：

1. 目标数据采集卡 IP 地址的设置
2. 采用 UDP 协议接收目标数据采集板数据
3. 解析数据并实时显示
4. 数据定时存储

为了保证数据采集系统的实时性，上位机软件采用多线程编程方式，初始化所需数据结构后开启数据读取线程与文件保存线程。

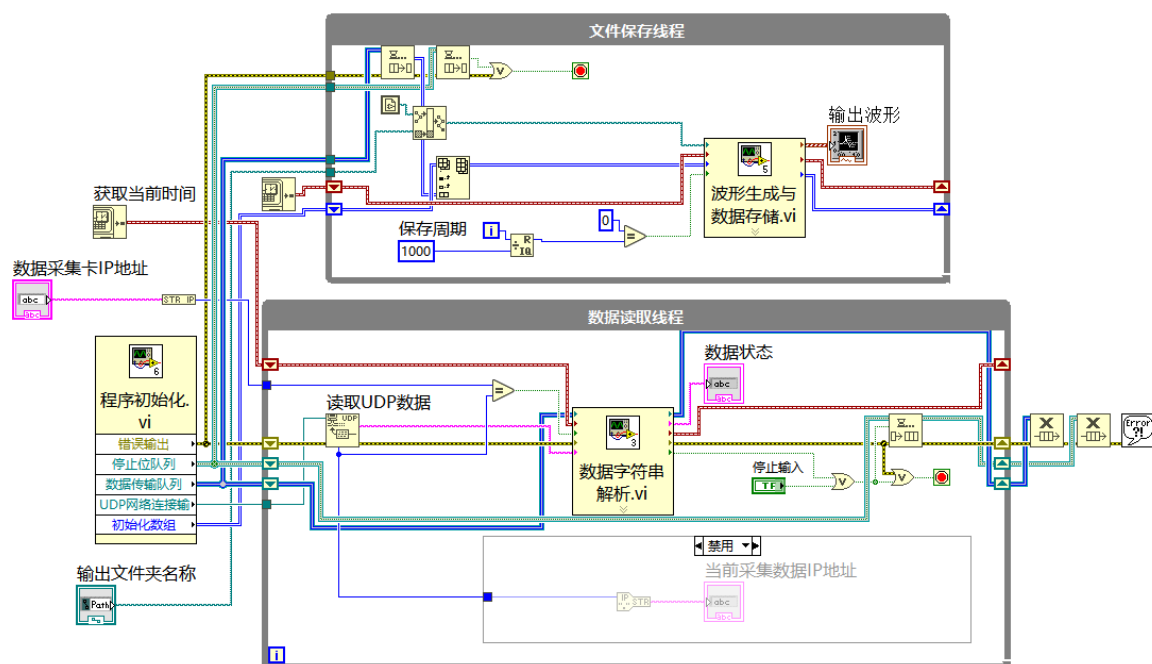


图 5-2 系统 LabView 主函数

“程序初始化”函数将创建两个队列，一路队列负责数据在两路线程间的传输；一路负责停止位的传输，保证能同时关闭两路线程。

5.3 数据读取线程

5.3.1 数据读取线程总循环

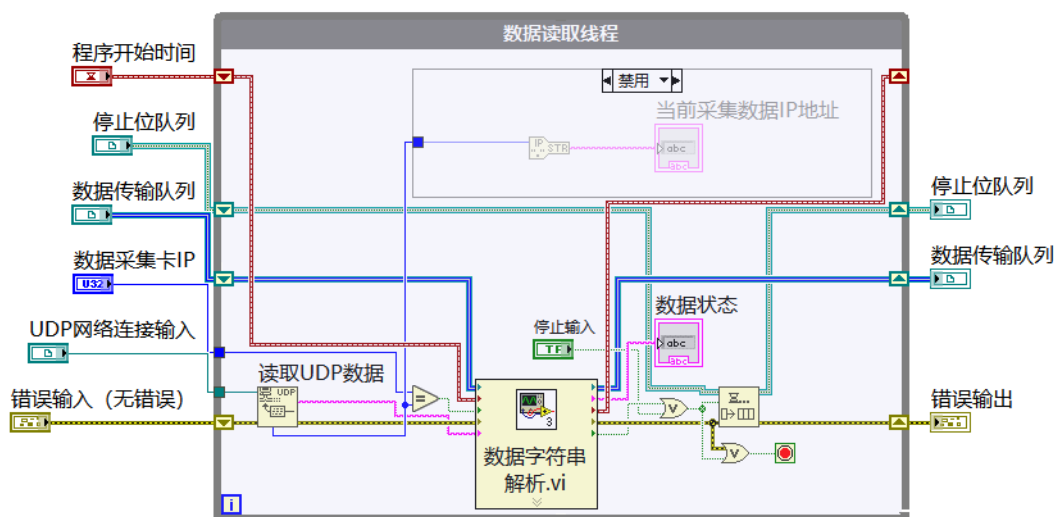


图 5-3 数据读取线程

LabView 并没有支持自制数据采集卡数据通讯的模块，系统采用 UDP 通讯并自行解包的方式实现上位机的数据接收功能。LabView 中 UDP 通讯模块将返回接收数据源的端口信息，将端

口信息与设置的目标 IP 地址进行比较，IP 地址正确则以字符串的形式返回来自下位机的数据并进行数据字符串解析操作，IP 地址错误则进行等待重新接收到来自指定 IP 地址的数据，直至超时报错退出程序。

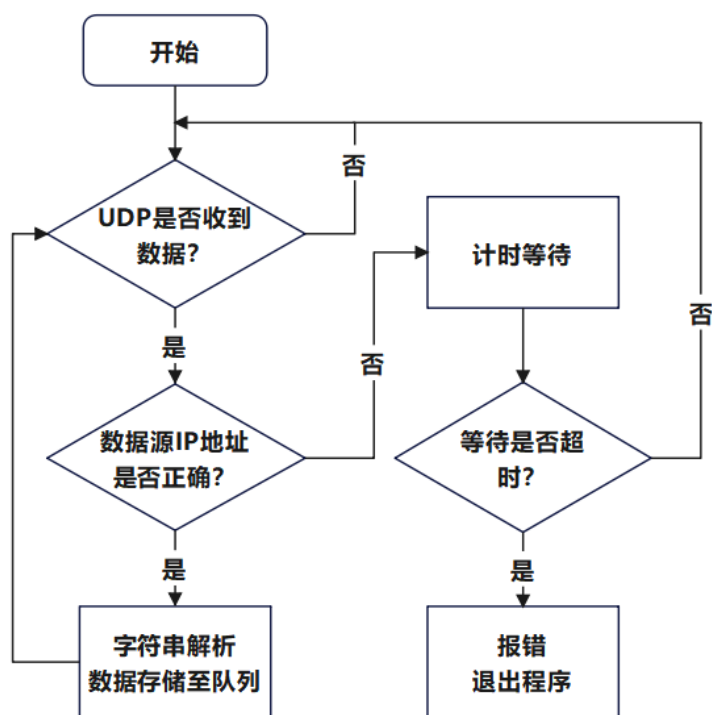


图 5-4 数据读取线程流程图

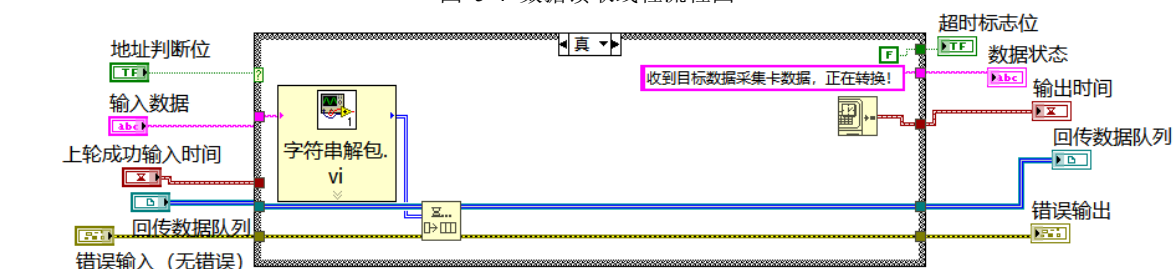


图 5-5 数据源 IP 地址正确时进行字符串解析

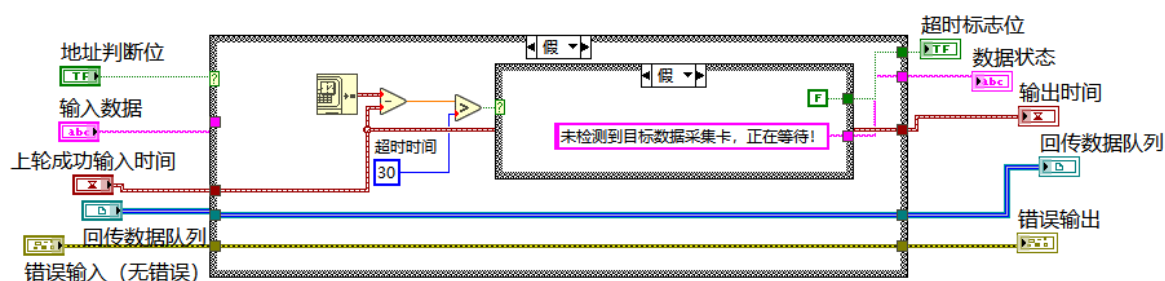


图 5-6 等待接收指定 IP 地址数据

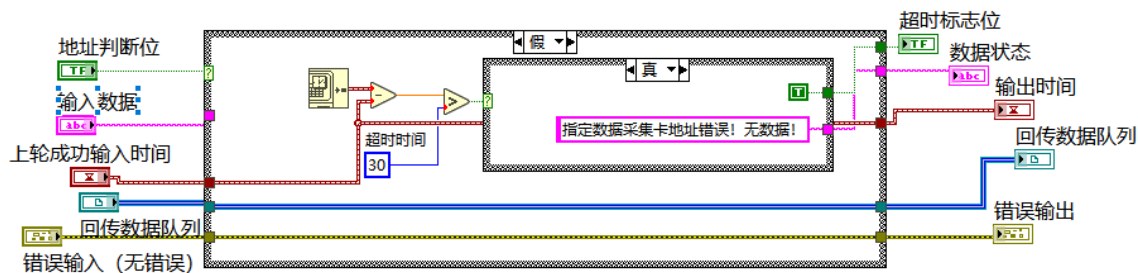


图 5-7 等待指定 IP 地址数据超时

5.3.2 字符串解包

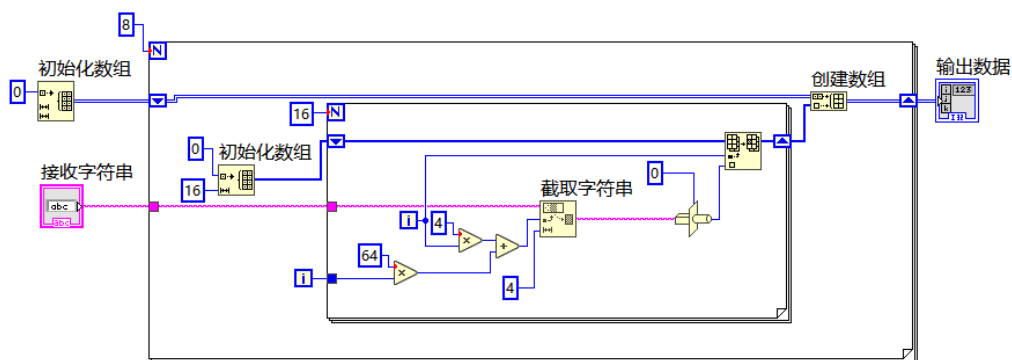


图 5-8 字符串解包

接收的字符串数据为 8 位无符号单字节整型数据，需转换至 32 位无符号长整型，且将各通道数据循环存至二维数组中。每次循环从字符串中取出单通道数据并存入内层移位寄存器中，16 通道数据提取完毕后存入外层移位寄存器中，内层移位寄存器清零并继续提取数据。

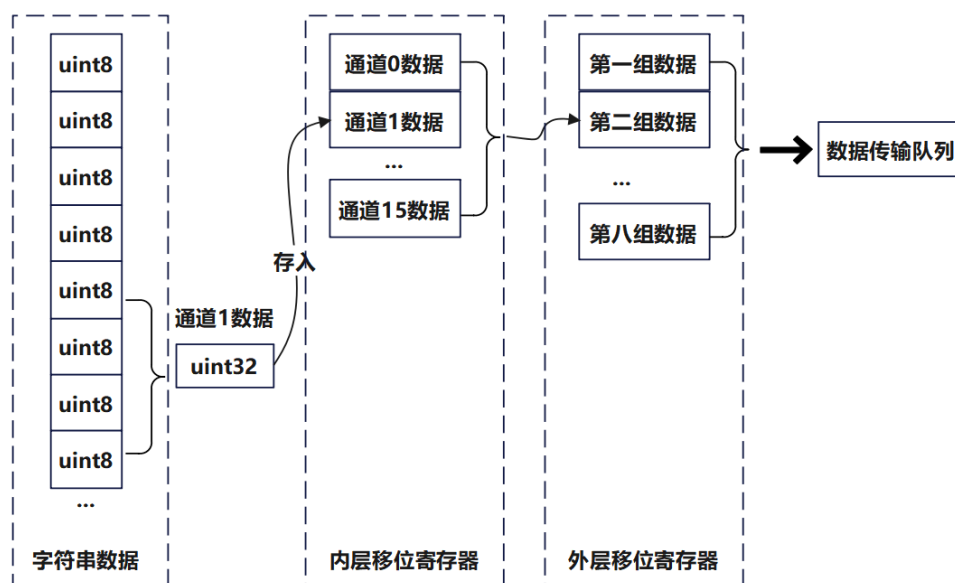


图 5-9 字符串解包流程图

5.4 文件保存线程

5.4.1 文件保存线程总循环

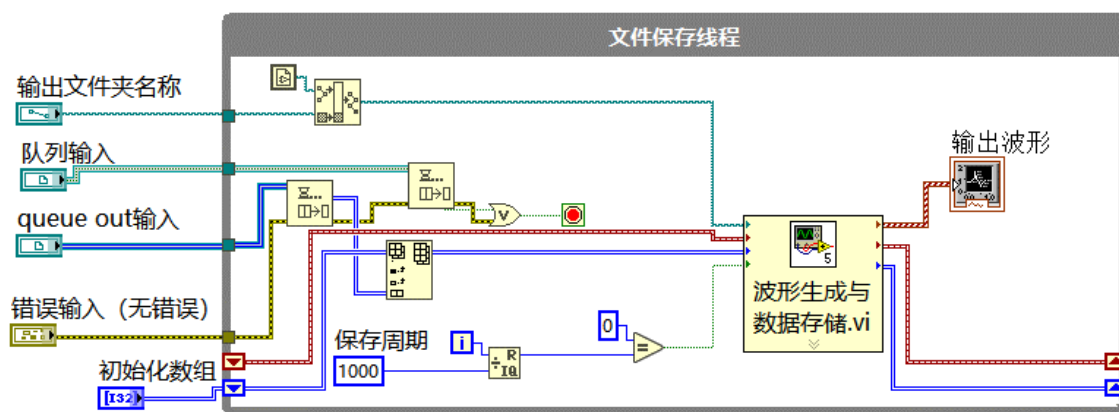


图 5-10 文件保存线程

文件保存线程接收数据读取线程中存入队列的数据、生成波形并存储。文件打开与写入需要耗费大量资源，程序运行时存在明显延迟，线程收到的数据采用移位寄存器的方式缓存，达到设定保存周期时触发波形生成与数据存储函数，将缓存数据写入文件中

5.4.2 波形生成与数据存储

当“文件保存线程”中所给“文件保存控制位”置真时，子函数将进行波形生成与数据存储。数据采集时不包含时间戳信息，上位机收到的数据仅为 ADC 在每次触发时扫描得到的电压值。ADC 采集的触发时间由定时器确定，为固定值，上位机可以自行添加时间戳以生成波形。

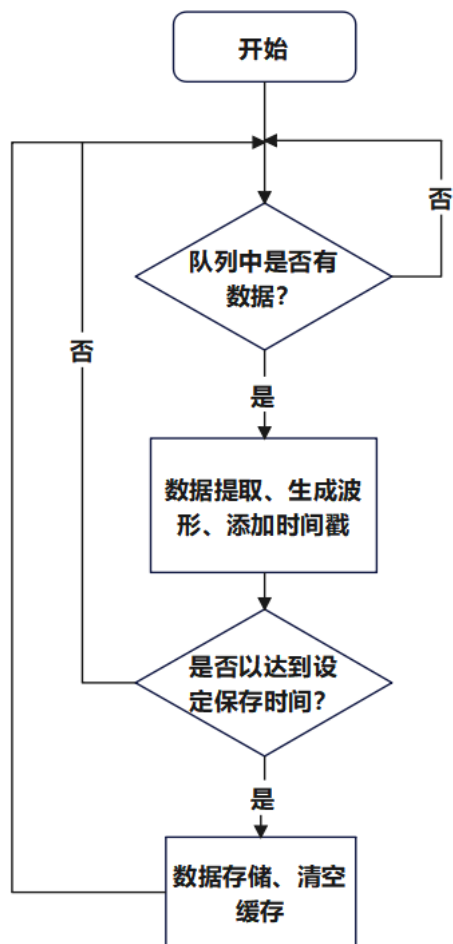


图 5-11 数据存储线程

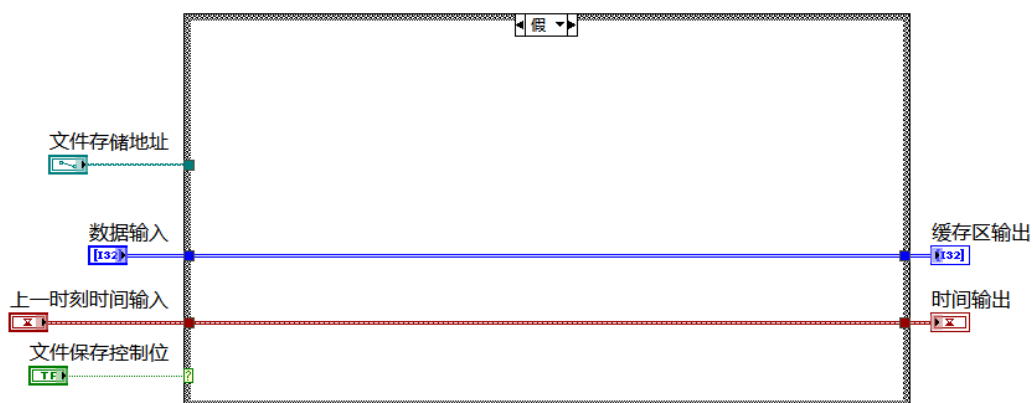


图 5-12 未达设定保存周期，不生成波形

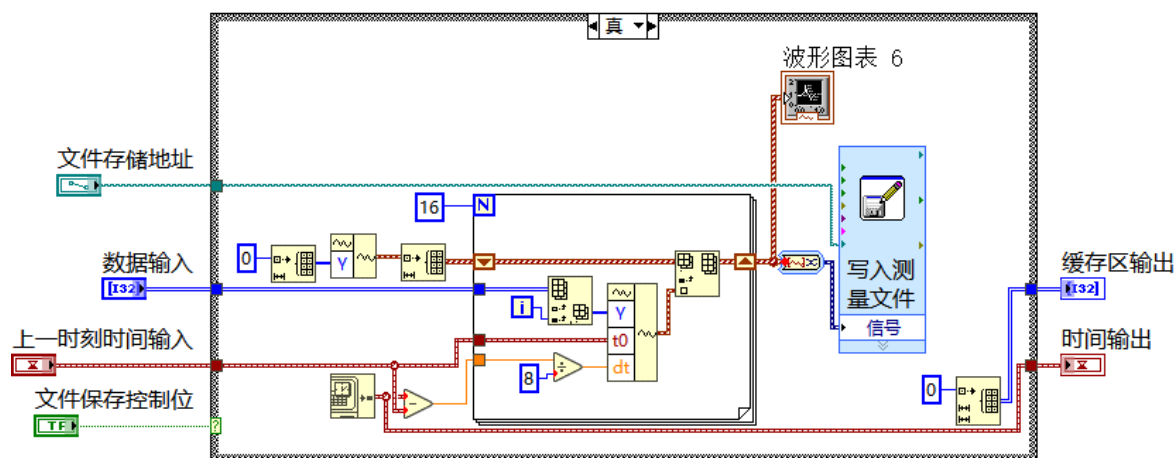


图 5-13 已达设定保存周期，进行波形生成与存储

此外，生成的波形可以调用 LabView 信号分析函数，实时显示其频谱图等特性。

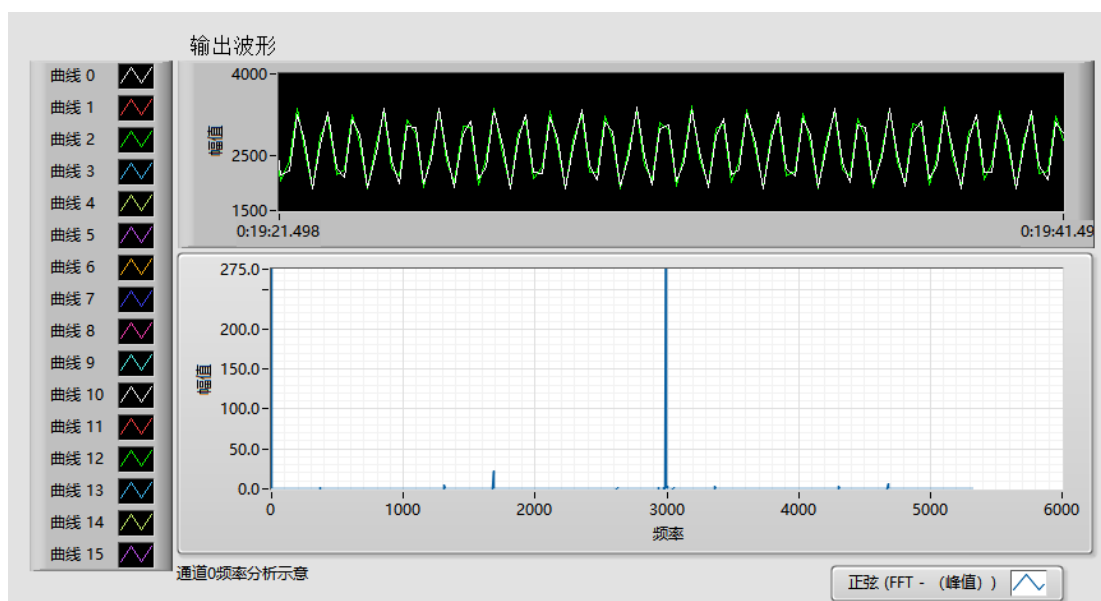


图 5-14 对 3kHz 正弦波进行简单频率分析

由于数据量较大，数据存储为 txt 格式文件排版混乱难以分析，csv 格式占用空间过大、读取缓慢，系统选择 TDMS 作为数据存储格式。TDMS 格式是 NI 公司推出的高速测试测量采集系统中的一种二进制数据存储类型，适合存储海量数据，兼有高度、方便和易存取等特点。选择存储波形与时间，设置每分钟保存子文件，最终存储的数据如下所示：

绝对时间	相对时间	通道0	通道1
2023/05/22 09:36:23.406 PM	0	2464	0
2023/05/22 09:36:23.507 PM	0.100871	2428	0
2023/05/22 09:36:23.608 PM	0.201742	2380	0
2023/05/22 09:36:23.709 PM	0.302613	2326	1
2023/05/22 09:36:23.810 PM	0.403484	2279	6
2023/05/22 09:36:23.910 PM	0.504355	2232	0
2023/05/22 09:36:24.011 PM	0.605225	2185	0
2023/05/22 09:36:24.112 PM	0.706096	2137	2
2023/05/22 09:36:24.213 PM	0.806967	2096	0
2023/05/22 09:36:24.314 PM	0.907838	2053	0
2023/05/22 09:36:24.415 PM	1.008709	2012	0
2023/05/22 09:36:24.516 PM	1.10958	1979	0
2023/05/22 09:36:24.617 PM	1.210451	1951	0
2023/05/22 09:36:24.717 PM	1.311322	1926	1
2023/05/22 09:36:24.818 PM	1.412193	1909	9
2023/05/22 09:36:24.919 PM	1.513064	1900	0
2023/05/22 09:36:25.020 PM	1.613935	1880	0
2023/05/22 09:36:25.121 PM	1.714805	1894	1
2023/05/22 09:36:25.222 PM	1.815676	1914	0

图 5-15 TDMS 数据

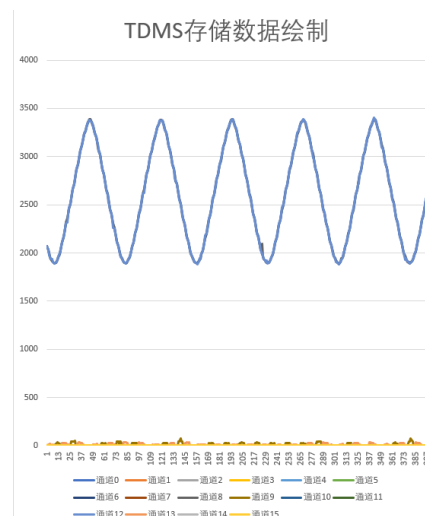


图 5-16 TDMS 存储数据绘制

6 系统测试与优化

系统设置为 10kHz 采样频率，接入 100Hz 正弦波信号进行采样测试，得到以下结果：

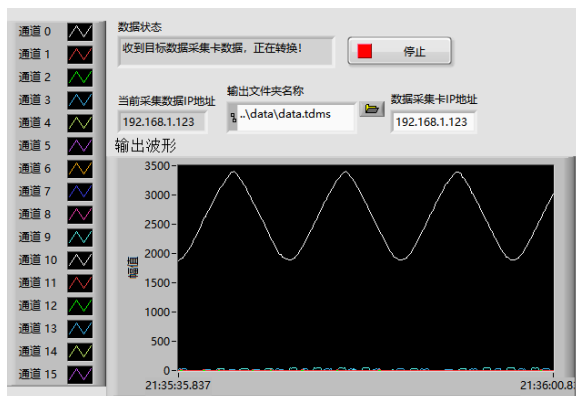


图 6-1 单通道 100Hz 正弦波信号采集

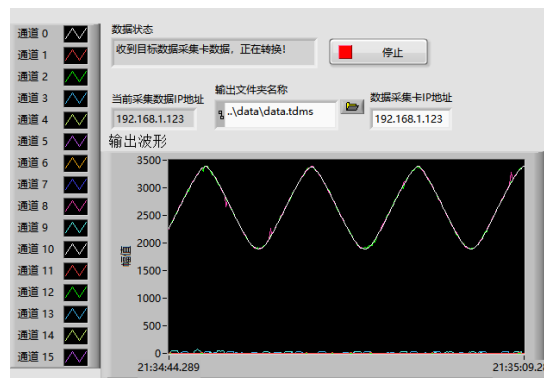


图 6-2 多通道 100Hz 正弦波信号采集

修改定时器参数，将 ADC 采样频率提高至 20kHz 后出现明显数据丢失问题：

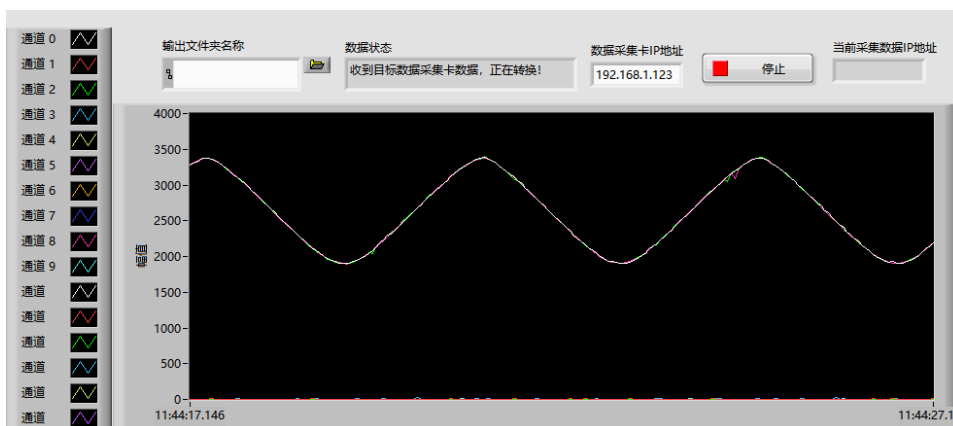


图 6-3 10kHz 采样频率下为完整波形

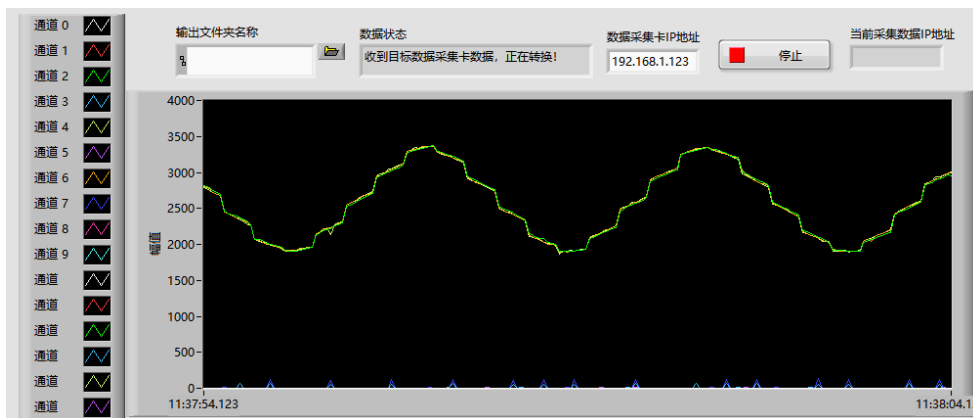


图 6-4 20kHz 采样频率下数据丢失

分析波形可知，所采集的正弦波信号变为锯齿状，且正好丢失一半数据。使用嵌入式系统程序开发软件 Keil 调试功能进行程序运行时间分析，经测试该功能存在较大问题，无法输出各段程序正确的运行耗时。采用系统定时器作为分析程序各部分操作耗时的依据，在进入各段程序前记录此时系统定时器的值，各段程序运行结束后再记录系统定时器值，相减可以得到各段程序运行所耗大致时间。

逐步分析程序各部分操作所耗时如下：

表格 6-1 20kHz 采样率下优化前程序各操作耗时

程序操作	耗时 (us)
数据打包与大小端转换	107
网络通信	1020

测量得 DMA 数据溢出时间如下，数据操作总耗时（即数据打包、网络通信总耗时）大于 DMA 数据溢出所用时间，DMA 中数据未经处理即被新数据覆盖。

表格 6-2 程序未优化前数据操作总耗时与 DMA 中断溢出耗时对比

程序操作	耗时 (us)
DMA 数据溢出耗时	409
数据操作总耗时	1127

分析可得，优化程序、提高数据采集频率有以下两种方式：

1. 延长 DMA 数据溢出耗时，降低对数据操作速率要求

在有限内存空间中最大化存入数据量。GD32 处理器板载 ADC 模块精度为 12 位，DMA 中仅用 16 位宽度内存即可满足数据存储要求。在 UDP 网络通信每次发送数据长度无法改变的前提下尽可能发更多组数据，优化数据结构后 DMA 数据溢出耗时明显上升，即对数据操作速率要求明显降低。

2. 降低数据操作耗时，提高数据处理速率

优化数据结构后，数据打包与大小端转换所需操作减半，耗时明显下降。网络通信中，使用 SPI 通信将数据发送至 W5500 芯片，网络通信耗时为限制采样频率的主要因素，采用以下两种方式进行优化：

1. 降低 SPI 时钟分频系数、提高 SPI 通讯时钟频率

2. 使用 DMA 模块进行 SPI 通讯

DMA 传输能大大缓解总线负担，减少总线时序建立时间，处理后的数据直接存入 DMA 中，由 DMA 模块负责将数据发至 W5500 网络模块。

表格 6-3 DMA0 部分通道请求表

通道 0	通道 1	通道 2
------	------	------

ADC0

SPI0_RX

SPI0_TX

GD32 芯片 DMA0 模块支持连接 ADC 与 SPI 外设，且 ADC0 模块与 SPI0 使用不同通道，允许同时工作。

修改相关函数配置，优化程序后得到以下数据：

表格 6-4 程序优化后 20kHz 采样率下各操作用时对比

程序操作	耗时 (us)
数据打包与大小端转换	69
网络通信	238

表格 6-5 程序优化后数据操作总耗时与 DMA 中断溢出耗时对比

程序操作	耗时 (us)
DMA 数据溢出耗时	798
数据操作总耗时	308

结合数据可知，DMA 数据溢出耗时明显上升，对数据操作速率要求明显下降；数据操作耗时明显下降，运行效率明显上升。优化后程序对 20kHz 正弦波信号采样如下：

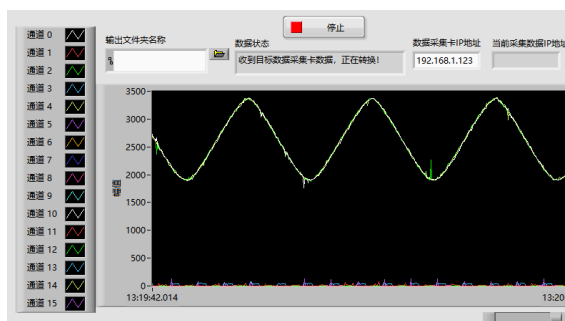


图 6-5 20kHz 采样频率下 100Hz 正弦波信号采集

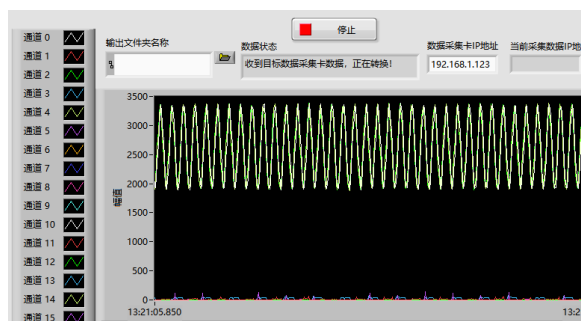


图 6-6 20kHz 采样频率下 3kHz 正弦波信号采集

分析操作耗时可得，数据采集系统理论上最高可在 40kHz 采样频率下稳定采集完整数据，测试得：

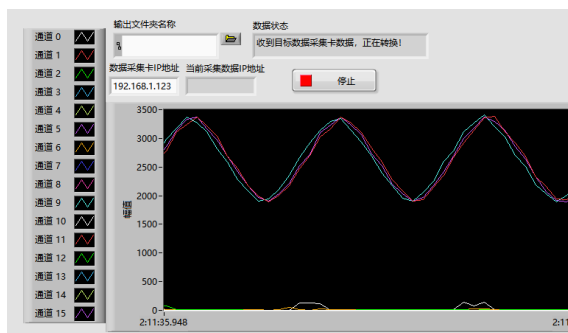


图 6-7 40kHz 采样频率下采集 3kHz 正弦波信号

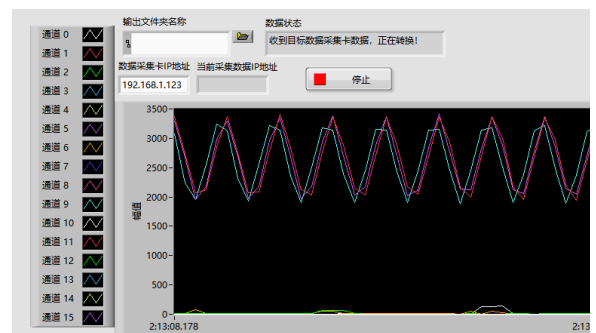


图 6-8 40kHz 采样频率下 8kHz 正弦波信号采集

程序连续运行 1 小时，共存储 19.48GB 数据，每分钟接收 2400000 条数据，即采样率为 40kHz，未发生断联、程序崩溃等问题，证明系统实现长时间内进行稳定数据采集、数据保存功能。

7 总结与展望

课题分析了目前已有数据采集方案，针对风力发电机运行维护时叶片相关数据的监测与采集问题，明确数据采集方案要求，提出风电运维多通道数据采集系统。主要完成工作与研究成果如下：

- （1） 基于 GD32 微处理器实现成本低廉、可靠的数据采集板总体软硬件方案设计与电路整体方案开发。数据采集板可同时连接 16 路传感器进行数据采集。使用 PoE 供电，数据采集板无需额外电源供电，仅需网线即可同时实现数据传输与供电，成本低廉且方便远距离部署于风机叶片各部位。
- （2） 实现对数据采集板嵌入式系统的开发，完成对多通道传感器高频数据采集、数据打包、数据发送等功能。在嵌入式系统上部署网络通信功能，基于 UDP 网络通信协议实现了远距离大量数据的高速传输。
- （3） 基于 LabView 完成数据采集系统上位机软件开发，自行编写仪器 I/O 模块，实现了对自制数据采集板进行数据接收、波形显示与数据存储的功能，并可对采集波形进行初步信号分析。
- （4） 完成了对数据采集系统的实际测试与优化，提高程序运行效率，将系统采样频率从 10kHz 提升至 40kHz，并在长时间测试中保持稳定，稳定采样频率持续记录采样数据，且数据所占空间较小，未出现程序崩溃、断联、采集数据丢失等问题。

系统仍存在一定不足，可作为下一步研究内容：

1. 系统采样频率不够高

限制数据采样频率的关键在于网络通信时长。W5500 网络通信模块支持 8 个独立端口同时通讯，用户可以将文件分块至各个端口，实现高速并行传输。GD32F303 微处理器为单核心处理器，硬件上无法实现多线程操作，无法发挥多个 Socket 并发高速传输的功能，仅能使用单端口通信，限制了网络通信速率。使用多核处理器可以提高系统采样频率。

2. 网络通信数据可能丢失

尽管 UDP 协议具有简单、传输速率快等特点，在实际测试中并未出现明显数据丢失问题，考虑到测试时长不足，数据采集系统应当尽量使用更为可靠的网络传输协议（如 TCP 协议）。

参考文献

- [1] J. Thorleifson, T. Davies, M. Black, D. Hopkin, and R. Verrall, "The Theseus autonomous underwater vehicle: A Canadian success story," in Proc.[J]IEEE/MTS Oceans Conference and Exhibition, 1997, pp. 1001–1008.
- [2] 罗承先.世界风力发电现状与前景预测[J].中外能源,2012,17(03):24-31.
- [3] 纪志国.我国风电产业现状与发展趋势探究[J].中国设备工程,2020,(18):217-218.
- [4] Larsen F M, Sorensen T . New lightning qualification test procedure for large wind turbine blades. 2003.
- [5] Galappaththi U I K , Silva A K M D , Macdonald M , et al. Review of inspection and quality control techniques for composite wind turbine blades[J]. Insight - Non-Destructive Testing and Condition Monitoring, 2017, 54(2):82-85.
- [6] I. Amenabar, A. Mendikute, A. López-Arraiza, M. Lizaranzu, J. Aurrekoetxea Comparison and analysis of non-destructive testing techniques suitable for delamination inspection in wind turbine blades,Composites Part B: Engineering,Volume 42, Issue 5,2011,Pages 1298-1305.
- [7] 张宇. 大型风力机叶片的振动分析与优化设计[D].沈阳工业大学,2013.
- [8] 王艳青. 复合材料风机叶片疲劳损伤监测研究[D].哈尔滨工业大学,2010.
- [9] Gad H E, Gad H E. Development of a new temperature data acquisition system for solar energy applications[J]. Renewable energy, 2015, 74: 337-343.
- [10] 张月,陶林伟.基于 FPGA 与 STM32 的多通道数据采集系统[J].西北工业大学学报,2020,38(02):351-358.
- [11] 李冰洋. 麦克风阵列声源定位系统设计与实现[D].电子科技大学,2020.
- [12] Eren, Halit & Potter, David. (2012). Data Acquisition Fundamentals. 10.1201/b11093-23.
- [13] 丁珍红. 基于 STM32 的高速大容量数据采集存储系统设计[D].上海交通大学,2011.
- [14] 陈仲生,杨拥民,杨光瑜.DI-720 系列高速数据采集卡及其应用[J].兵工自动化,2002,(01):55-57.
- [15] 孟武胜,朱剑波,黄鸿,赵晨光.基于 LabVIEW 数据采集系统的设计[J].电子测量技术,2008,(11):63-65
- [16] 迟钦河,赵仲生,乔桂芳等.89C51 单片机在多通道数据采集系统中的应用[J].自动化仪表,2000(06):33-35+39.DOI:10.16086/j.cnki.issn1000-0380.2000.06.012.
- [17] 李红刚,张素萍.基于单片机和 LabVIEW 的多路数据采集系统设计[J].国外电子测量技术,2014,33(04):62-67.

- [18] 韦顺.芯片：产能吃紧 价格普遍上涨[J].股市动态分析,2020,No.1507(24):50-51.
- [19] 刘青正,庞晓明.串行通信隔离方法应用[J].中国科技信息,2018,(20):77-78.
- [20] 谢希仁.计算机网络[M].5版.北京;电子工业出版社,2005.XIE Xiren.Computer network[M].5th ed.Beijing:Electronic Industry Press,2005.
- [21] 胡志华,郭其一.基于 IEEE802.3af 的以太网供电技术(PoE)[J].仪表技术,2007,(04):54-56.
- [22] IEEE802.3af Power-over-Ethernet TI solutions for both ends of the cable[OL].White Paper of Texas Instru-ments.
- [23] 吴迪等编著;张大波主编;.嵌入式系统原理、设计与应用[M].机械工业出版社,2005.
- [24] 李会兰.嵌入式 Internet 互联网技术的应用及研究[J].长江信息通信,2022,35(02):103-105.
- [25] 王力生,梅岩,曹南洋.轻量级嵌入式 TCP/IP 协议栈的设计[J].计算机工程,2007,No.270(02):246-248.
- [26] 毕爱波.基于 TCP/IP 协议的嵌入式 Internet 技术的研究与应用[D].中国海洋大学,2005.
- [27] TOMIYAMA H.CHIKADA S.HON DAS.An RTOS based Approach to Design and Validation of Embedded System[A.IEEE Intemational Symposium on VLSI Design, Automation and Test C].2005.185-187
- [28] 刘滨,王琦,刘丽丽.嵌入式操作系统 FreeRTOS 的原理与实现[J].单片机与嵌入式系统应用,2005,(07):8-11.

谢 辞

在此，我想感谢同济大学机械学院的朱玉田老师，感谢徐州市建筑设计院的陆爱中老师在学生毕业设计开展中提供了细心的指导与宝贵建议。

此外，我想感谢同济大学开展的德语强化班，修读德语的一年我度过了美好的时光，与许多不同专业的同学结下深厚友谊，也遇到了我的两位优秀的舍友山明扬与潘元皓同学，是二位的帮助让我摆脱浑浑噩噩的状态，从人生一系列重大打击中走出来。

我想感谢同济大学 SuperPower 机器人战队，感谢战队的指导老师何俊杰老师，战队中有太多的故事值得多年后细细品味。那种面对惨败的不甘、悔恨，那种不是为了证明什么，只是纯粹的想把它做好的信念，支撑着我在数不清多少个日夜里研发、工作。我还要感谢靠谱的队友们陪我一块犯傻，作为一个团队共同为同一个信念奋斗。21 年惨败的时候，我拿走了写着“挫折”的胸牌；23 年惜败的时候，我拿的是“极致”。我不再认为这是场挫折，不再有遗憾，不再有悔恨，我知道我已尽己所能。下一次，我们会做得更好。回首本科生涯，我可以自豪地说，我没有虚度光阴。

再后，我还想感谢学校、学院和本科期间的所有任课老师对我的栽培，感谢他们的授课，拓宽了我的知识面，启发我选择了感兴趣的研究方向。

最后，我想感谢我的家人，正是他们在我本科期间对我无微不至的关怀、支持我做的每一个重要决定，才使得我能无所顾忌地发展自己的爱好，追求自己的理想。