

iHIP手持式数据采集仪

—产品技术方案

[版本号：V1.0.0]

西安因联信息科技有限公司

**目 录**

[1 技术选型 3](#_Toc502133194)

[2 技术实现方式 3](#_Toc502133195)

[2.1 手持设备硬件技术实现 3](#_Toc502133196)

[2.1.1 主板设计 3](#_Toc502133197)

[2.1.1.1 电源 3](#_Toc502133198)

[2.1.1.2 LCD 4](#_Toc502133199)

[2.1.1.3 按键 5](#_Toc502133200)

[2.1.1.4 WiFi 7](#_Toc502133201)

[2.1.1.5 光线 8](#_Toc502133202)

[2.1.1.6 LED 8](#_Toc502133203)

[2.1.1.7 蜂鸣器 9](#_Toc502133204)

[2.1.1.8 RTC 9](#_Toc502133205)

[2.1.1.9 SD卡 9](#_Toc502133206)

[2.1.1.10 Uart 10](#_Toc502133207)

[2.1.1.11 USB 10](#_Toc502133208)

[2.1.2 采集板设计 11](#_Toc502133209)

[2.1.2.1 电源模块 11](#_Toc502133210)

[2.1.2.2 采集模块 12](#_Toc502133211)

[2.1.2.3 控制模块 13](#_Toc502133212)

[2.1.3 板间通信问题 13](#_Toc502133213)

[2.2 手持设备软件技术实现 13](#_Toc502133214)

[2.2.1 数据采集 14](#_Toc502133215)

[2.2.1.1 振动采集 14](#_Toc502133216)

[2.2.1.2 压力采集 15](#_Toc502133217)

[2.2.2 按键声音、电源指示灯、按键背光灯 16](#_Toc502133218)

[2.2.2.1 按键声音 16](#_Toc502133219)

[2.2.2.2 电源指示灯、按键背光灯 17](#_Toc502133220)

[2.2.3 波形控件实现方式说明 17](#_Toc502133221)

[2.3 上位机软件技术实现 17](#_Toc502133222)

[2.3.1 上位机技术架构 18](#_Toc502133223)

[2.3.2 数据库实现技术 18](#_Toc502133224)

[2.3.3 数据共享接口 18](#_Toc502133225)

[2.3.4 数据分析 18](#_Toc502133226)

[2.3.5 与手持设备通讯 19](#_Toc502133227)

[3 所选用开发工具及版本 19](#_Toc502133228)

[3.1 上位机开发环境 19](#_Toc502133229)

[3.2 手持APP开发环境 19](#_Toc502133230)

[3.3 手持底层开发环境 20](#_Toc502133231)

1. 技术选型

手持设备依据点检计划或临时任务，通过振动和压力传感器采集设备振动信号和压力信号，采用数字信号处理元件和技术，对采集信号进行实时计算和处理，输出被监测设备监测单元的加速度、速度、位移振动信号和压力信号，并以实际波形和数值形式显示，应用内以专业算法，提供时域、频域、总值趋势图、特征值分析谱图或工具，实现被监测设备异常事件的快速监测和智能状态评估。可通过USB方式、WIFI方式，实现手持设备与上位机系统通信，实现数据回收和进一步分析。

Android系统具有开放性，智能化，发展迅速，市场份额大，支持硬件丰富，功能强大等优势，是智能设备操作系统的很好选择。iHIP手持数据采集仪选择基于Android原生系统开发，将需要定制开发的6款APP作为Android的系统APP开发并安装使用。系统架构以launcher为第一应用，作为需求应用的列表中心和入口，此设计方便开发调试，后期维护和进程管理，同时能减少交互层级，减少复杂度。为以后有新需求功能扩展提供预留平台，也可根据业务关联程度设计开发隐藏的公共模块作为本地数据交互中心。

1. 技术实现方式
   1. 手持设备硬件技术实现
      1. 主板设计
         1. 电源

电源模块设计方案为：

* 充电芯片：美信Max8903------2A单节锂电池DC-DC充电器
* 电量监测芯片：美信Max17048-----小尺寸、微功耗锂离子电池电量计
* 电源管理芯片：飞思卡尔PF0100---14通道可配置电源管理IC (PMIC)



图1 手持仪器硬件平台电源模块框图

具体电源树如图2所示：



图2手持仪器硬件平台电源模块电源树

* + - 1. LCD

核心板自带LCD接口，但需要转接成所选LCD的线序。核心板LCD的数据位为888模式，所选LCD的数据位为666模式，本次设计为666模式，预留565模式。其连接关系如图3所示：



图3手持仪器硬件平台LCD模块示意图

选用3.5寸屏幕的其具体参数如表1所示：

|  |  |  |
| --- | --- | --- |
| 序号 | 项目 | 规格特性 |
| 1 | LCD Size | 3.5 inch(Diagonal) |
| 2 | Resolution | 480(horizontal)× 640(Vertical) |
| 3 | Interface | 18bit-RGB |
| 4 | Connect type | Connector |
| 5 | Color Depth | 18 bit |
| 6 | Technology type | a-Si |
| 7 | Display colors | 262K |
| 8 | Pixel Configuration | R.G.B. Vertical Stripe |
| 9 | Display Mode | Normally White |
| 10 | Surface Treatment | HC |
| 11 | Viewing Direction | 9 O’clock |
| 12 | LCM (W x H x D) (mm) | 64(H)\*85(V)\*2.9(T) |
| 13 | Active Area(mm) | 53.28(H) x 71.04(V) |
| 14 | LED Numbers | 8 LEDs |

* + - 1. 按键

按键模块的设计选用了TI的TCA8418，它是一款集成 ESD 保护的键盘扫描器件，18个GPIO可配置为8个输入和10个输出，以支持8x10的键盘阵列。其待机电流消耗可低至3uA。其与CPU连接关系如图4所示：



图4 手持仪器硬件平台Key模块示意图

<1> Power键的设计具有以下功能：

* 每次按下有红色led灯指示；
* 开机：按1秒可以开机；
* 开机后屏幕点亮时，短按可以熄屏；
* 开机后屏幕熄屏时，短按可以唤醒；
* 开机后屏幕点亮时，长按约2秒，屏幕上提示关机，用功能键选择关机；
* 死机时，长按大于5秒，由硬件强制关机。

<2> 扩展按键：

扩展按键根据仪器键盘提示，具有F1~F5功能键、选择键、删除键、确认键和数字键等。

<3> BOOT按键：

主板上有一个BOOT按键，用于系统烧写。需要烧写系统时，先按下BOOT键，再按电源键开机，直接进入烧写模式，使用USB线和MfgTool2工具操作。

<3> RESET按键：

调试过程使用，当需要复位的时候，可以按此键对整个系统进行复位操作。

* + - 1. WiFi

Wifi模块选用台湾正基 (AMPAK) 公司的AP6441三合一模块，模块内部为博通的BCM43341芯片组，实测WiFi吞吐量70Mbps，信道带宽HT20/40。此模块支持WiFi、蓝牙、NFC功能。其与CPU的连接关系如图5所示：



图5 手持仪器硬件平台wifi/NFC模块连接示意图

AP6441模块的功能参数具体如下：

WLAN：

* 双频带2.4/5GHz 802.11a/b/g/n
* 单流IEEE 802.11n支持20MHz和40MHz带宽，PHY层速率可达150Mbps；
* WLAN接口：SDIO v2.0 ------可达50 MHz时钟速率；
* 支持单天线共享WLAN和Bluetooth；

Bluetooth：

* 蓝牙4.0(BLE)低功耗；
* 蓝牙一类二类传输运行；
* 蓝牙接口：UART—可达4 Mbps，支持所有BT V4.0类型；
* 全面支持蓝牙省电模式(休眠和深度睡眠模式)；
* ECI—增强共存技术，支持WLAN接收时BT SCO的传输；

NFC：

* 读写模式；
* 主动和被动Peer-to-Peer(P2P)模式；
* 标签/卡仿真模式，对于swp\_0/swp\_1双UICC SIM卡接口，支持双单线协议(SWP)；
  + - 1. 光线

光线感应模块的设计选用了intersil公司的isl29023，它是一款集成检测自然光和红外光的数字转换器，其先进的自校准光电二极管阵列可以模拟人眼，具有优良的红外抑制作用。其工作电流最大85uA，待机电流最大为0.3uA，因此，特别适合用于对功耗比较敏感的手持产品中。其与CPU连接关系如图6所示：



图6 手持仪器硬件平台光线感应模块连接示意图

* + - 1. LED

LED模块选用三个2x3x4mm的方形高亮LED，分别为红、黄、绿色，主要用来指示电池的状态。其与CPU连接关系如图7所示：



图7 手持仪器硬件平台LED连接示意图

充电过程：

* 黄色：准备充电
* 红色：正在充电
* 绿色：充电完成

使用过程：

* 绿色：电量充电
* 黄色：电量不足
* 红色：严重不足，请接适配器
  + - 1. 蜂鸣器

蜂鸣器选用电磁式分体有源蜂鸣器：YMD-12065-G 3V（HXD），主要用来指示按键声音，其与CPU连接关系如图8所示：



图8 手持仪器硬件平台蜂鸣器连接示意图

CPU输出的PWM3经过缓冲门后控制集成负载开关的ON/OFF脚，从而打开或关闭电磁式分体有源蜂鸣器的的供电，达到蜂鸣效果。其中，集成负载开关FDC6330L是一个NMOS控制PMOS器件。

* + - 1. RTC

CPU自身带有RTC功能，但实测其功耗偏大，本次设计选用外部RTC芯片：intersil公司的isl1208，这是一款低功耗产片，特别适用于手持仪器。这款芯片支持双电源，仪器上电后，用VDD供电，仪器掉电后，用VBAT供电。当使用VDD供电时，其耗流只有1.2~4uA；当使用VBAT(纽扣电池)供电时，其耗流只有400~950nA。其与CPU连接关系如图9所示：



图9 手持仪器硬件平台RTC连接示意图

* + - 1. SD卡

核心板的存储器为eMMC，其容量为8G，专用于存储系统文件和采集数据。为了防止其不够用，本次设计扩展了SD卡，最大支持64G，接口为SDIO，其与CPU连接关系如图10所示：



图10 手持仪器硬件平台SD卡连接示意图

为了防止仪器摔落导致Micro SD卡跌出卡座，在布局设计中，卡座位置尽量靠近仪器边沿，后期会在其顶部点凝胶，沾上一块泡沫，缓冲仪器坠落的冲击力。

* + - 1. Uart

imx6q共有5路串口，其支持三种模式：串行RS-232NRZ模式、9位RS-485模式、IrDA模式(接口)。其与CPU连接关系如图11所示：



图11 手持仪器硬件平台SD卡连接示意图

核心板将5路串口全部引出，其用途为：

* Uart1：2线调试串口，经过232转换后引出到J5接口；
* Uart2：2线引出串口，经过232转换后引出到J4接口；
* Uart3：4线串口，用作与蓝牙模块的通讯接口；
* Uart4：4线串口，引出到采集板，用其复选功能作为SPI1接口；
* Uart5：4线串口，引出到采集板，用其复选功能作为SPI1接口。
  + - 1. USB

imx6q共有2路USB口，分别为：OTG USB和Host USB。本次设计选用OTG USB作为外部访问接口，通过外接USB线，实现从电脑端访问仪器内部存储器。其与CPU连接关系如图12所示：



图12 手持仪器硬件平台USB连接示意图

OTG USB支持主从设备自动检测功能，当检测自己做主设备时，CPU通过USB\_OTG\_PWR\_EN使能供电电路，由主板给外部设备供电(+5V)，当检测自己做从设备时，则不使能供电电路，由外部设备供电。

* + 1. 采集板设计



图13手持仪器硬件平台采集板整体框图

采集板分为电源模块、控制模块、采集模块。

* + - 1. 电源模块



图14手持仪器硬件平台采集板电源模块框图

电源模块由主板直接提供4.2V电压，上电使能受主板CPU控制，电源模块实现将4.2V转变为采集模块所需的各种电压，包括：+24V、±6V、±5V、+3.3V、+2.5V、+1.8V、+1.2V等电压。

CycloneIV系列FPGA，CPU需要：3.3V(高有效)、2.5V(低有效)、1.2V(高有效)三路IO使能，无上电次序，一起使能。

* + - 1. 采集模块



图16 振动采集模块框图

采集模块主要由前级滤波电路、信号增益衰减电路、转差分电路、硬件积分电路、A/D转换电路等组成。实现将传感器的输出信号进行调理后给FPGA处理。

为适应不同的传感器调理，需要通过数字开关或者固态继电器来改变信号走向。SSR为固态继电器，SW为数字开关。

* + - 1. 控制模块



图18 控制模块框图

控制模块主要通过一个FPGA实现，FPGA芯片通过SPI1接收主板CPU的指令，从而配置采集模块的参数，例如ADC的工作模式、采样率、积分选择等；同时，FPGA芯片还要把采集到的压力或振动数据进行处理后通过SPI2发送给主板CPU，再经过软件算法处理后进行展示和存储。

* + 1. 板间通信问题

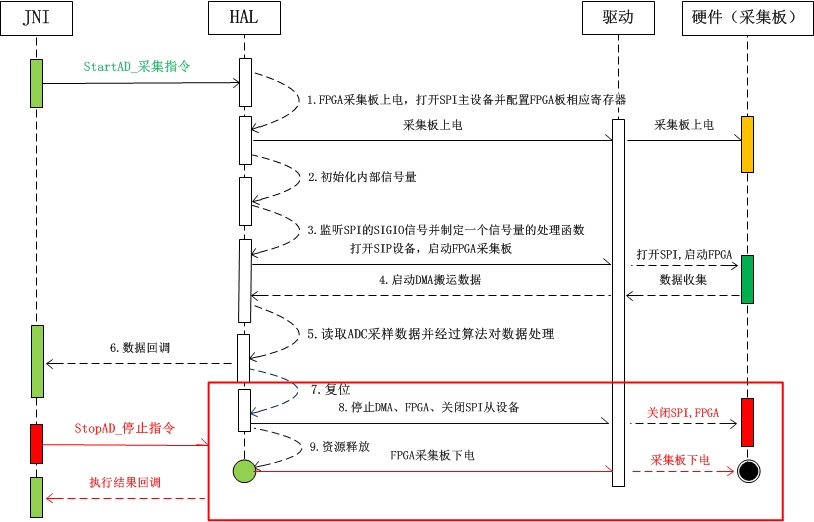


图19 主板与采集板接口部分框图

如图19所示，主板与采集板通过SPI1、SPI2和Power Ctrl接口进行通信。

* SPI1用于读写FPGA内部的寄存器和下发相关指令，配置相关采集参数并获得采集板工作状态 ，以及开启和关闭采集状态。
* SPI2用于上传采集的压力数据和振动数据。
* Power Ctrl主要用于管控采集板的供电，包含：+4.2V供电、3.3V\_EN、2.5V\_EN、1.2V\_EN。
  1. 手持设备软件技术实现
     1. 数据采集

手持设备采集模块主要分为两大类型：压力采集、振动采集，其中压力采集又分为实时压力和压力标零，而振动采集又分为单通道分析和振动等级评估。以下是采集模块设计方案架构图：



采集模块架构继续沿用Android原生框架，根据用户需要扩展相应功能。

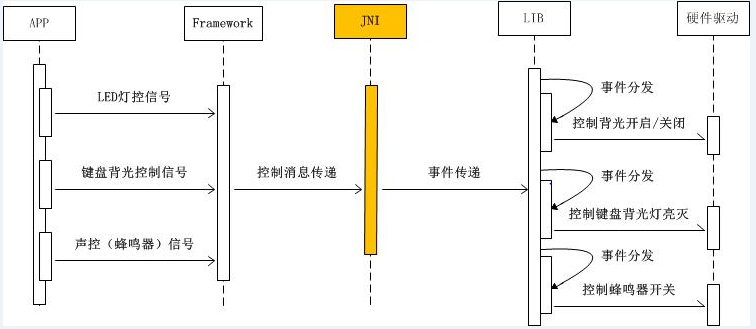
JNI层：用C++代码封装HAL层的本地函数，向framework层提供调用的接口。

HAL层：向JNI层提供调用的接口，响应JNI层的传递的事件（如开始采集，停止采集），并主动回调最终结果数据给JNI层。

硬件驱动层：驱动底层硬件模块，并根据HAL层下发的指令控制实现相应的功能。

* + - 1. 振动采集
* 开始采集：
  + 首先给FPGA采集板上电，打开SPI主设备并配置FPGA板相应寄存器。
  + 初始化内部信号量。
  + 捕捉SPI底层驱动发送的SIGIO信号并指定一个信号处理函数用于读取相应ADC采样数据。
  + 打开SPI从设备，并启动DMA搬运数据，启动FPGA采集板。
  + 当接收到指定长度数据后退出采集进行各种数据运算，最终将运算结果主动回调给JNI层。
  + 停止DMA搬运数据，停止FPGA采集板，关闭SPI从设备，释放资源，
* 停止采集：
  + 停止DMA搬运数据，停止FPGA采集板，关闭SPI从设备，关闭SPI主设备，FPGA采集板下电，释放资源，最终将停止状态主动回调给JNI层。
    - 1. 压力采集
* 压力标零操作：
  + 不接压力传感器，根据采集配置参数，空气中空采一组ADC采样数据，并求出其平均值，并以此为基准。
* 压力数据采集：
  + 接上压力传感器，根据采集配置参数获取一组ADC采样数据。
  + 判断采样数据和标零值差值若小于0.5bar,则表示这组数据无效，并提示用户。
  + 判断采样数据和标零值差值若大于0.5bar,则表示这组数据有效，并计算相应的表盘数据或压力波形数据。
* 压力表盘算法逻辑如下：
  + 对有效数据求平均值
* 压力波形数据算法逻辑如下：
  + 对有效数据先进行FFT运算，对运算后数据求出前3个最大值及对应的index值，求出index值最小值，进而求出周期内的点数。
  + 对有效数据求最小值对应的index值。
  + 以有效数据最小值为起点，以周期内点数分段，分别向左向右对多个分段求最大值最小值，后对所有分段内的最大值最小值再求平均，以此为波形数据最大最小值。
  + 以有效数据最小值为起点，向右取一个周期内点数个数据为最终波形显示数据。
    1. 按键声音、电源指示灯、按键背光灯

按键声音和电源指示灯及按键背光灯三个附加功能主要都基于Android原生架构所做的定制开发，主要有APP层、framework层、JNI层、HAL层、硬件驱动层，具体架构图如下：



APP层：系统设置app更改设备状态。

Framework层：监听设备状态，调用相应JNI。

JNI层：用C++代码封装HAL层的本地函数，向framework层提供调用的接口。

HAL层：响应JNI层的传递的事件，并向JNI层提供调用的接口。

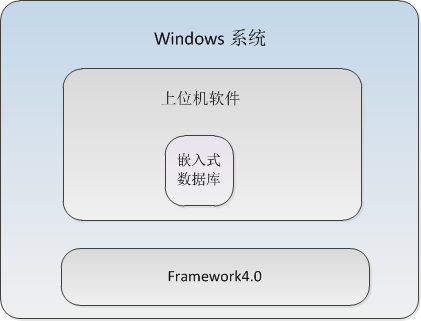
硬件驱动层：根据HAL层的指令控制实现相应的功能。

* + - 1. 按键声音
* 开按键声音：
  + 系统设置app打开开关，更改状态。
  + framework监听状态，调用JNI。
  + JNI层打开HAL层设备文件。
  + JNI层下发打开按键声音指令给HAL层。
  + HAL层打开Kernel层按键响应设备文件和声控设备文件，持续监测按键响应事件，若监测到有按键按下，则设置声音占空比让蜂鸣器发声15ms后自动关闭发声。
* 关按键声音：
  + 系统设置app打开开关，更改状态。
  + framework监听状态，调用JNI。
  + JNI层下发的关闭按键声音指令给HAL层。
  + HAL层设置kernel层声音占空比让蜂鸣器停止发声，关闭按键响应和声控两个设备文件。
    - 1. 电源指示灯、按键背光灯
* 开启：
  + 系统设置app打开开关，更改状态。
  + framework监听状态，调用JNI。
  + JNI层打开HAL层设备文件。
  + JNI 层下发开指示灯指令给HAL层。
  + HAL层打开Kernel层电源指示灯驱动设备文件，控制底层驱动对应指示灯的GPIO脚为高电平。
* 关闭：
  + 系统设置app打开开关，更改状态。
  + framework监听状态，调用JNI。
  + JNI层下发关指示灯指令给HAL层。
  + HAL层根据JNI层要关闭的指示灯，从而控制底层对应指示灯的GPIO脚为低电平。
  + 关闭HAL层设备文件。
    1. 波形控件实现方式说明

波形控件基于开源第三方图表库MPAndroidChart二次开发实现，通过向图表控件传入采集数据信息，做为图表数据源，实现采集信息波形绘制、展示功能。通过更新图表数据源，实现采集波形的缩放、平移操作。通过记录、更新游标参考线位置，采用图表控件自带的标记功能，实现游标功能。

* 1. 上位机软件技术实现
     1. 上位机技术架构

采用.Net Framework 4.0框架，以Windows 窗体应用的方式实现上位机软件架构，上位机为单用户桌面应用程序，软件技术架构如下图：



* + 1. 数据库实现技术

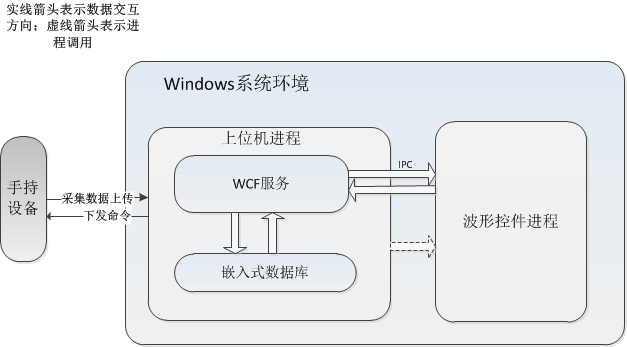
上位机数据存储使用嵌入式数据库Firebird作为存储数据库，方便安装部署，数据访问接口使用.Net类库 FirebirdSql.Data.FireBirdClient.dll提供访问Firebird数据库的通用接口。

* + 1. 数据共享接口

数据接口采用WCF REST部署方式，将服务寄宿于上位机软件运行进程中，数据接口以标准HTTP请求形式进行访问，提供JSON格式的数据，因而不限制调用方的实现技术及系统平台类型，符合SOA规范。

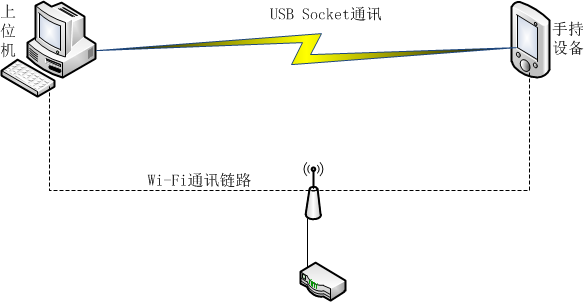
* + 1. 数据分析

数据分析模块使用Labview作为技术工具来完成，以独立进程的方式提供给上位机进程进行调用，上位机进程与数据分析模块进程之间通过WCF命名管道方式（IPC地址）进行数据交换。



* + 1. 与手持设备通讯

通讯模块支持USB和Wi-Fi无线两种通讯方式，且通讯数据遵从TCP/IP协议，以Socket数据包的形式进行数据交换。通讯链路图如下：



1. 所选用开发工具及版本
   1. 上位机开发环境

* **开发工具**：Microsoft Visual studio 2012
* **运行环境**：.Net Framework 4.0
* **数据库**：Firebird 2.5
  1. 手持APP开发环境
* **开发环境**：
  + JDK 1.8.0\_131
  + Android SDK4.4,API 19
* **开发工具**：Android Studio 2.3
  1. 手持底层开发环境
* **开发工具：**Microsoft Visual C++
* **运行环境：**Ubuntu 12.04