

2020年TI杯 福建省大学生电子设计竞赛

目录

1	方案综述	2
2	理论分析与计算	3
2.1	体表温度测量方法	3
2.2	LMT70 温度算法	3
2.3	运动量统计	3
3	各模块简介	4
3.1	Arduino 心电数据数据处理层	4
3.2	ESP32	4
3.3	后端服务器层	5
3.4	前后端沟通层	5
3.5	前端显示层	5
	附录	6
A	测试数据	6
A.1	心率测试	6
A.2	温度测试	6
A.3	运动统计测试	6

摘要

本作品由 ADS1292R 模块, LMT70 测温模块, MPU6050 加速度计步检测模块, 再结合 Arduino 和 ESP32 系列 MCU 和基于 Node.js 的服务器组成。

关键词: Node.js、物联网、心率检测

1 方案综述

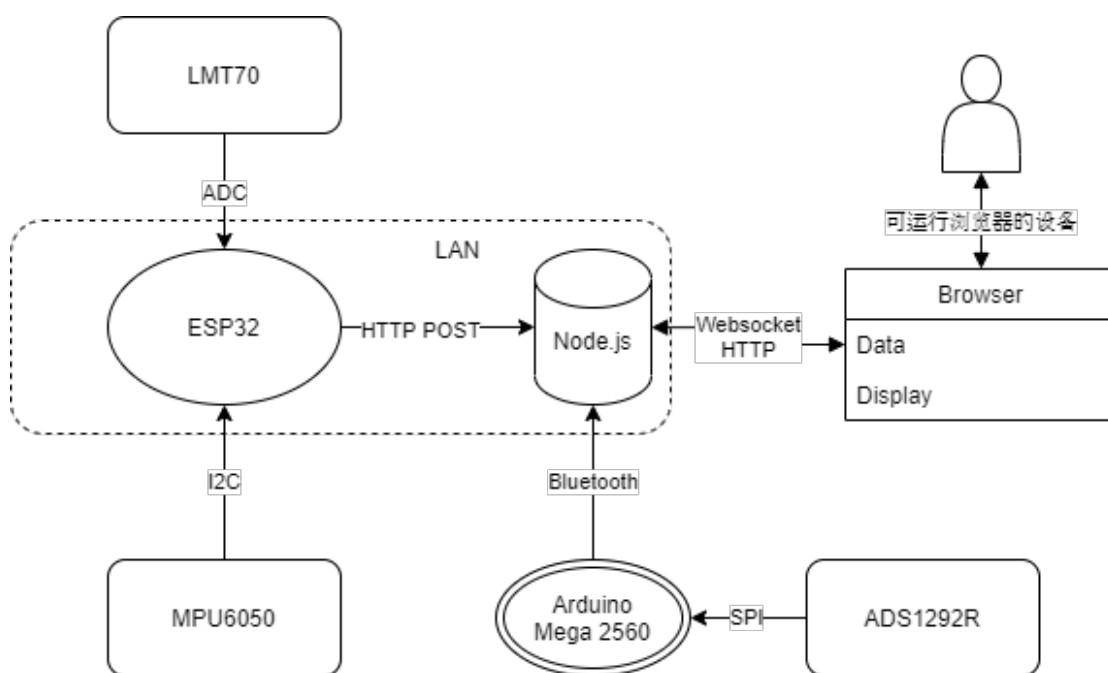


图 1: 系统框图

根据题目要求可得到系统的总体框图, 如图 1 所示。

人体产生的心电信号经过 ADS1292R 模块的处理后, 产生含有少量噪声的心电信号, 传输给 Arduino Mega 2560 板处理得到心率数据。Arduino 再通过蓝牙模块直接传送数据到 Node.js 服务器。

与此同时 MPU6050 加速度计步检测模块通过对人体行进时 x, y, z 三轴加速度的变化计算得到人行进时的步数, 并且 LMT70 芯片对空气中温度变化敏感, 而引起的芯片输出电压的变化。LMT70 测温模块、MPU6050 加速度检测模块与 ESP32 直接相连, 将收集到的数据进行处理后以 JSON 格式通过 HTTP 协议传输到本地服务器。最后, Node.js 服务器后端利用 Websocket 协议将数据处理后发送给前端页面显示。用户可以使用浏览器接收到所需的数据。

ESP32、Arduino 和执行 Node.js 的服务器终端必须要求在同一局域网之内, 其中有多种类型的设备可以运行 Node.js 作为服务器, 如路由器, 树莓派等。本次作品采用笔记本电脑作为 Node.js 的服务器端设备进行操作。

2 理论分析与计算

2.1 体表温度测量方法

采用归一化加权平均算法，每进行一次测量，对人体进行多次采样，得到温度的测量数列，首先得到拥有一致性的测量数据，然后依据本算法得出融合值，进而算出温度的准确测量结果，去除过程中的不确定性

由一致性数组 $X \in [1, N]$ 能够得出被测数据的平均值

$$\bar{X} = \frac{1}{N} \sum_{i \in [1, N]}^m X_i \quad (1)$$

计算每一测量值 X_i 相对于均值 \bar{X} 的偏差量 ΔX_i

$$\Delta X_i = X_i - \bar{X} \quad (2)$$

权值函数

$$f(\Delta X_i) = \left(\frac{1}{\pi * \Delta X_i} \right) * \arctan\left(\frac{1}{\Delta X_i}\right) \quad (3)$$

将偏差量 ΔX_i 代入权值函数 $f(X)$ ，作归一化处理 $\Delta \bar{a}_i$

$$\Delta \bar{a}_i = f(\Delta X_i) \quad (4)$$

由归一化偏差量得到加权值 \bar{a}_i

$$\bar{a}_i = \frac{\Delta \bar{a}_i}{\sum_{i \in [1, N]} \Delta \bar{a}_i} \quad (5)$$

由加权值得到最终的平均值 X^+

$$X^+ = \sum_{i \in [1, N]} X_i \times \bar{a}_i \quad (6)$$

2.2 LMT70 温度算法

查 LMT70 的数据表易知，可以使用一阶传递函数来计算 LMT70 所感测的温度，但是在较宽的温度范围内，它是最不准确的方法。我们使用在电气特性温度查找表 (LUT) 中找到的 LUT(查找表) 信息可以轻松生成方程式。其二阶传递函数为

$$T_M = a \cdot (V_{TAO})^2 + b \cdot (V_{TAO}) + c \quad (7)$$

更加精确的三阶传递函数为

$$T_M = a \cdot (V_{TAO})^3 + b \cdot (V_{TAO})^2 + c \cdot V_{TAO} + c \quad (8)$$

2.3 运动量统计

采用波峰波谷检测法。人体迈步过程中，重心会随着人体运动在一定范围内出现规律性的变化。脚蹬地离开地面时，地面的反作用力会使垂直加速度开始增大，身体重心上移，当脚达到最高

位置时, 脚的垂直速度最小, 但垂直加速度最大。当脚向下落时, 垂直加速度开始减小, 落地时加速度达到最小值。在运动过程中, 加速度传感器会根据人体运动的频率采集到 x, y, z 三轴的加速度数值, 形成一个波形图。在波形图中存在波峰和波谷, 检测步数其实就是检测波峰。行走时, 由加速度传感器获取的数值经过检测如果是波峰, 并且两次波峰的时间以及阈值符合设定的数值范围, 那么即可算作用户走了一步, 超过这个设定的范围则看作用户停止步行或者无效行走。

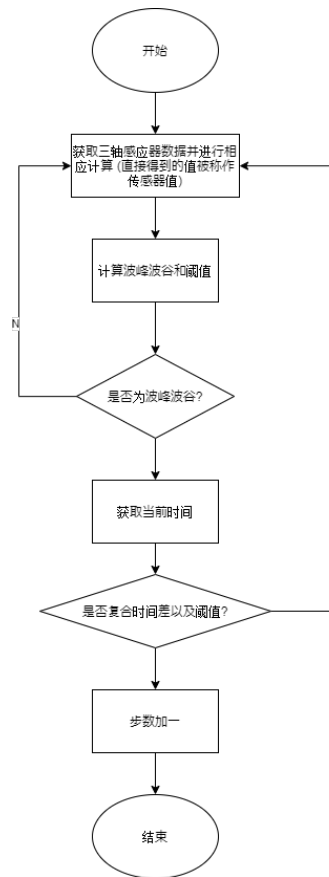


图 2: 算法流程图

3 各模块简介

3.1 Arduino 心电数据数据处理层

ADS1292 的数据会经过 Arduino 的外接的 HC-05 蓝牙模块与 Node.js 服务器进行沟通。Arduino 是一款便捷灵活、方便上手的开源硬件产品, 其标准化的接口模式为它的可持续发展奠定了坚实的基础。

3.2 ESP32

ESP32 带有蓝牙与 WiFi 模块, 这意味着你可以通过 Wi-Fi 或蓝牙以极低的价格轻松地远程控制 and 监控设备。但由于其 GPIO 引脚口数目有限, 使得其无法直接接入 ADS1292 模块。但

是 ESP32 可以方便的采集处理以及传输来自 LMT70 和 MPU6050 的数据.

3.3 后端服务器层

使用 Node.js 和 HTTP 协议作为服务器层的处理架构. 之所以使用 HTTP 而非 MQTT 作为沟通协议是因为其以 TCP 为传输层的特点, 保证了数据的完整发送和接收. 其易于操作与实现, 且在各个终端设备均可使用, 不局限于移动端或者桌面端.

Node.js 是一个 Javascript 运行环境. Node.js 使用事件驱动, 非阻塞 I/O 模型而得以轻量和高效. 当 Node.js 执行 I/O 操作时 (例如从网络读取、访问数据库或文件系统), Node.js 会在响应返回时恢复操作, 而不是阻塞线程并浪费 CPU 循环等待. 非常适合于处理大量终端设备连接时的情形. 其强大的扩展功能亦保证了蓝牙模块的正确接入.

3.4 前后端沟通层

WebSocket 的诞生本质上就是为了解决 HTTP 协议本身的单向性问题: 请求必须由客户端向服务端发起, 然后服务端进行响应. 比起传统的 Ajax 来说更为简练明了.

3.5 前端显示层

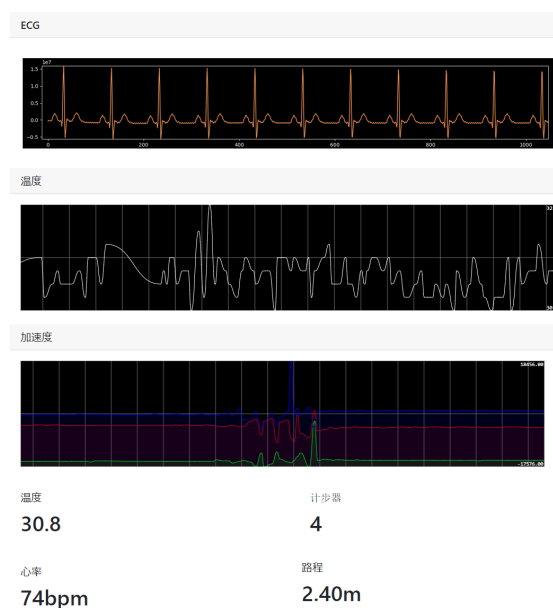


图 3: 前端页面测试效果

因为 Web 只需要页面开发, 没有开发语言的限制也没有客户端和服务端的限制问题, 这样就会使成本大大降低. Web 是可以跨平台的, 这样就节省了下载安装的时间, 也不会占用手机内存和手机流量. 同时也有很好的展示效果.

附录

A 测试数据

A.1 心率测试

芯片测得的心率	手环测得的心率	误差
87	88	1.15%
86	88	2.33%
87	87	0.00%

表 1: 心率测试精度统计

A.2 温度测试

芯片测得的温度	测温枪测得的温度	误差度数	误差
35.00	34.40	0.6	1.71%
34.70	35.00	0.3	0.86%
34.80	34.60	0.2	0.57%

表 2: 温度测量统计

A.3 运动统计测试

芯片测得的步数	手环测得的步数	误差
10.00	14.00	0.00%
15.00	16.00	6.67%
30.00	31.00	3.33%

表 3: 计步器功能统计