

简易尿常规检查替代仪器

李佳俊；李大林；方任斌

第一部分 设计概述

1.1 设计目的

尿常规检查是根据病人提供的尿液，通过机器检验以及人工操作对尿液进行分离检查，测定尿液中各种成分的量是否异常，以此帮助疾病的诊断。但在医院进行尿常规检查流程复杂，对于行动不便而需要体检的人群造成了极大困扰。为了解决该问题，团队制作了简易的尿常规检查替代仪器。仪器组成包括：数据采集部分、信号接收电路、结果分析处理终端。使用该仪器，可便捷地实现对尿液中蛋白质、葡萄糖、酮体（丙酮）等成分含量测定，通过博流板卡计算分析并利用无线子系统在手机端实时显示，对疾病的诊断有良好的初筛功能。

1.2 应用领域

本作品属于家用医疗器械领域，对于行动不便而又需要去医院进行尿常规检查的人群，可以通过本仪器做初筛功能，或者在日常生活中，该仪器可用于定期的简易尿常规体检。

1.3 主要技术特点

主控制器采用博流公司的 BL618 开发板，辅以拓展显示屏用于检测结果的展示。对于数据的处理，分为两个部分：硬件部分是多路激光通道和高精度偏振谐波检测系统，软件部分通过线性回归模型建立数据分析算法与诊断模型，依照专业药液相色谱质谱分析结果作为对比训练组进行算法的修正。对于检测结果，可在本地显示屏与手机端同时查看。

1.4 关键性能指标

- (1)、验证精度高：误差小于 0.02(g/100ml)；
- (2)、检测周期短，一轮检测只需不到五秒即可显示结果；
- (3)、所建模型判断准确率高，不会发生待测物体种类检测错误的情况。

1.5 主要创新点

- (1)、方案创新，利用谐波检测的方式，配合硬件软件协同计算，确保检测结果的准确性。
- (2)、应用创新，抓住特殊家庭医疗过程的痛点，帮助人们便捷地完成某些疾病的初筛。
- (3)、参考论文《A Fast-Response Ultraviolet Phototransistor with a PVK QDs/ZnO Nanowire Heterostructure and Its Application in Pharmaceutical Solute Detection》中提出的对于液体溶质的二次谐波检测方案，进行改进、应用。
- (4)、模型创新，建立线性回归模型并训练数据集，最终得到标准模型。

第二部分 系统组成及功能说明

2.1 整体介绍

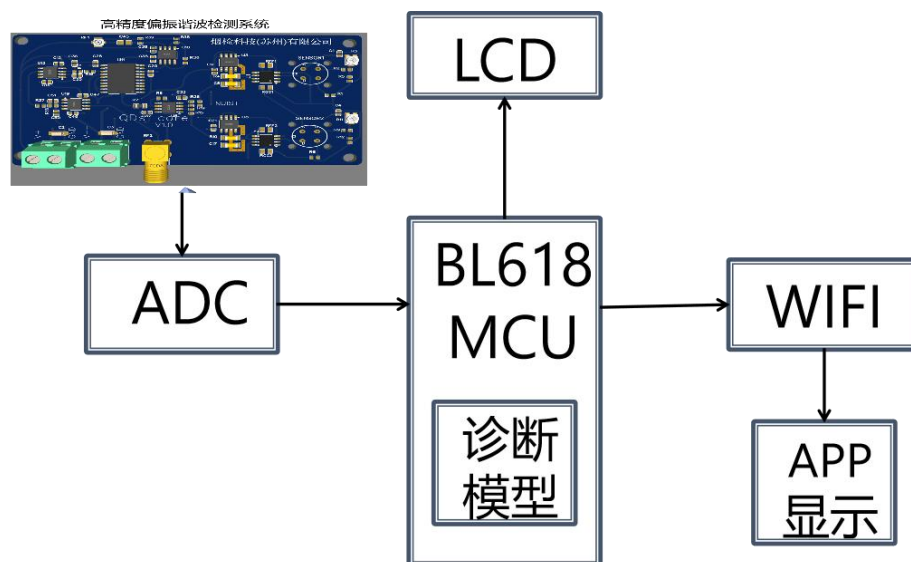


图 1 系统框图

系统框图如图 1 所示，主控采用博流公司的 BL618 主板。ADC 采集高精度偏振谐波检测系统产生的二次谐波信号，用于分析判断成分。LCD 显示屏用于显示检测结果，包括待测成分含量状态（正常为 negative 阴性，异常为+++）。同时，显示结果包括该种情形下可能的病症。WIFI 模块用于连接 onenet，通过 HTTP 协议将数据上传到云端，将检测的数据结果同时发送给手机端，实现便捷实时查看。

2.2 各模块介绍

2.2.1 多路激光通道

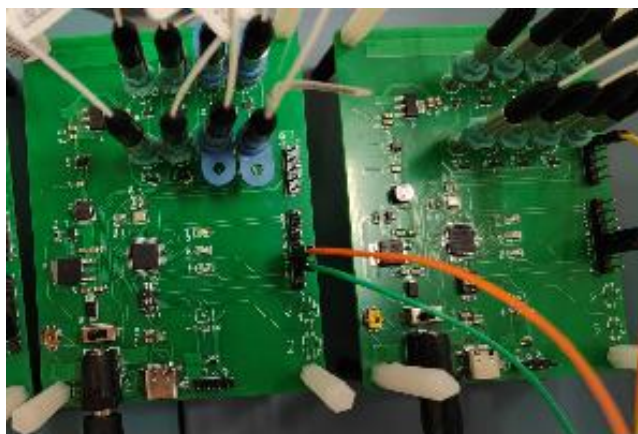


图 2 多路激光通道实物图

本系统的多路激光通道包括依次电性连接的电源处理模块、多路可调谐激光通道模块、光电转换模块。其中电源处理模块为多路可调谐激光通道模块提供稳定的工作电压，多路可调谐激光通道模块的光源为轮流循环输出，且光源输出的信号顺序以及光源轮流循环输出的速度可以调节，而光电转换模块将激光通道输

出的光源信号转换成电信号。

电源处理模块包括电源输入端口、电源开关、稳压芯片、稳压二极管、线性稳压器、降压电路。由于激光通道的工作电压为 1.5V，所以需要处理对输入的电源。首先，稳压二极管稳定电路因为电流原因发生较大的电压变化；然后连接线性稳压管，其中，线性稳压管具体为 L78M05CDT-TR，作用是消除与单点调节相关的分配问题，并将电压稳定在 5V，线性稳压管后接电感去除直流电中可能存在的一些交流电；进一步的，连接降压电路 LDO，具体为 ME6211C33M5G-N，将电压稳定在 3.3V；最终接稳压芯片，具体为 AMS1117，将 3.3V 电压稳定在 1.5V，达到激光通道的工作电压。

2.22 高精度偏振谐波检测系统

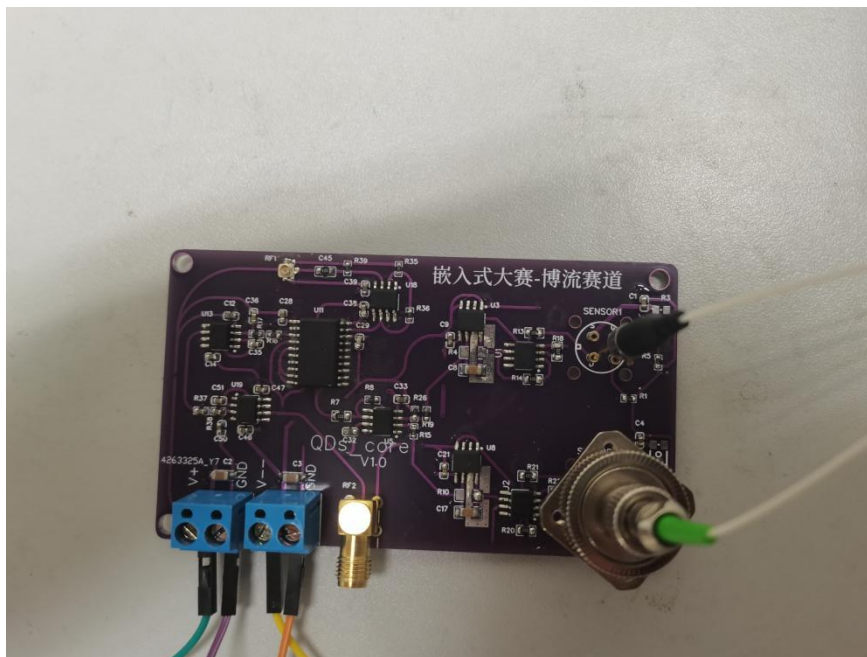


图 3 高精度偏振谐波检测系统实物图

高精度偏振谐波检测系统是硬件系统中最主要的部分。首先介绍朗伯比尔定律：当一束平行单色光垂直通过某一均匀非散射的吸光物质时,其吸光度 A 与吸光物质的浓度 c 及吸收层厚度 b 成正比,而与透光度 T 成反相关。这就说明,当溶液中溶质浓度发生变化时,该溶液吸光度也会发生变化。基于此原理,我们设计了该硬件系统。该系统核心是差分放大电路,差分输入一端接入待测溶液,另一端接入标准溶液。当相同的激光分别射入两种溶液后,被吸收后的光由光纤接入该系统,先由光信号转换为电信号,然后再进行差分放大。为了结果便于分析,我们将所得差分信号进行调制,得到二次谐波信号,从而进行分析。如图 4 所示,该流程图更加直观地展现了分析过程。

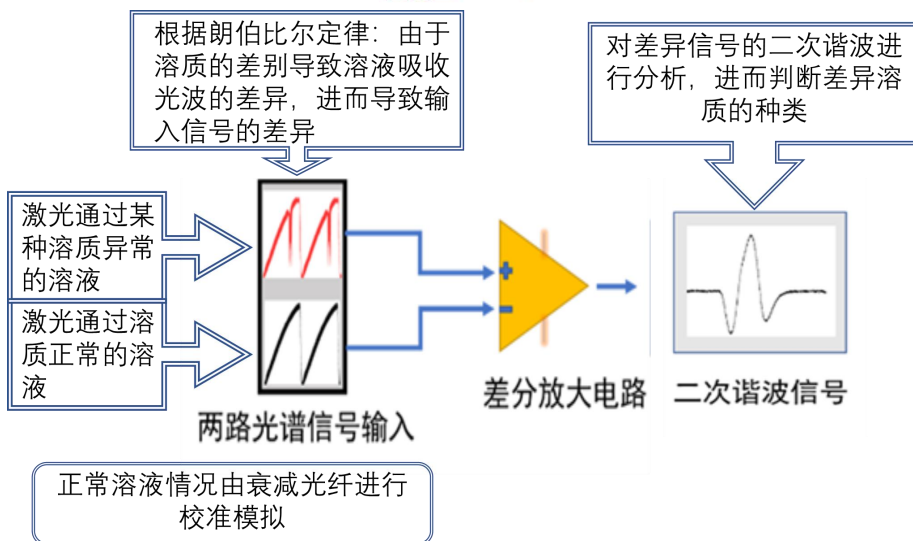


图 4 高精度偏振谐波检测系统实现过程

2.23 BL618

作为系统主控，并实现软件系统的设计（如图二）。首先，待测成分通过硬件系统得到二次谐波信号（电压数据），将这些数据分为算法训练组和算法测试组，通过图五所示框图进行数据集的训练，从而得到标准模型。其中，算法使用的是线性回归模型，代码在附录中。

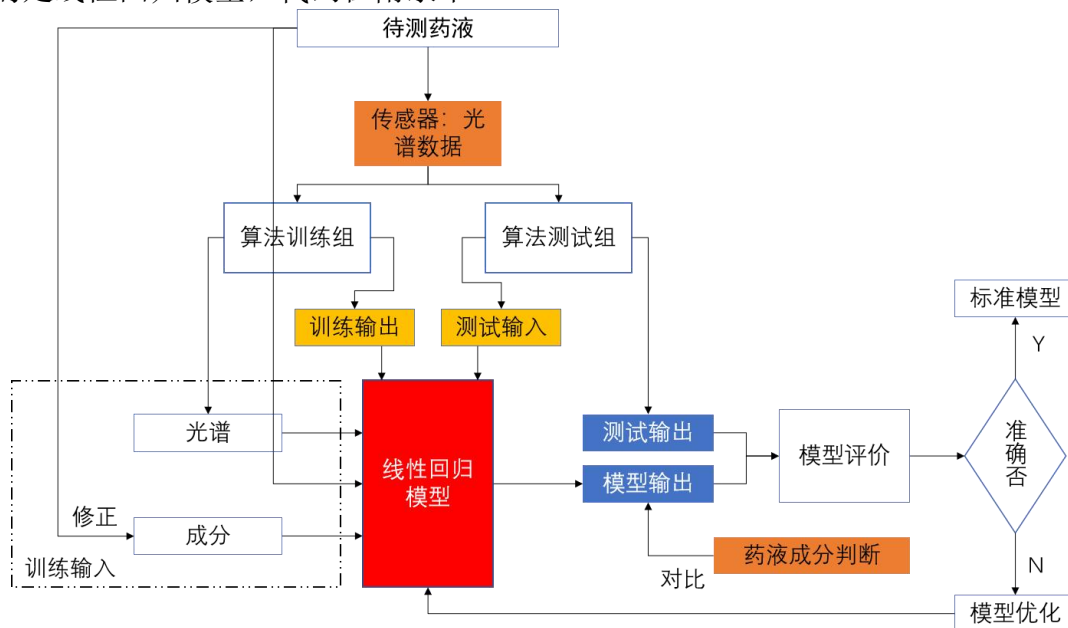


图 5 软件设计框图

2.24 LCD 模块

通过 LCD 显示屏来展现最后的检测结果，该系统选取的三个待测成分分别是葡萄糖，酮体（丙酮）和蛋白质。检测结果与标准值对比，判断成分是否超标，并提醒体检人员该种状况下可能发生的疾病。

2.25 WIFI 模块

通过 WIFI 连接 Onenet 平台。Onenet 是由中国移动打造的 PaaS 物联网开放

平台，平台能够帮助开发者轻松实现设备接入与设备连接，快速完成产品开发部署，为智能硬件、智能家居产品提供完善的物联网解决方案。最后在手机端同步显示结果。

2.26 ADC 模块

利用 ADC 模块接收硬件系统的二次谐波信号，用于软件系统中的数据处理。

第三部分 完成情况及性能参数

3.1 外部其他硬件展示

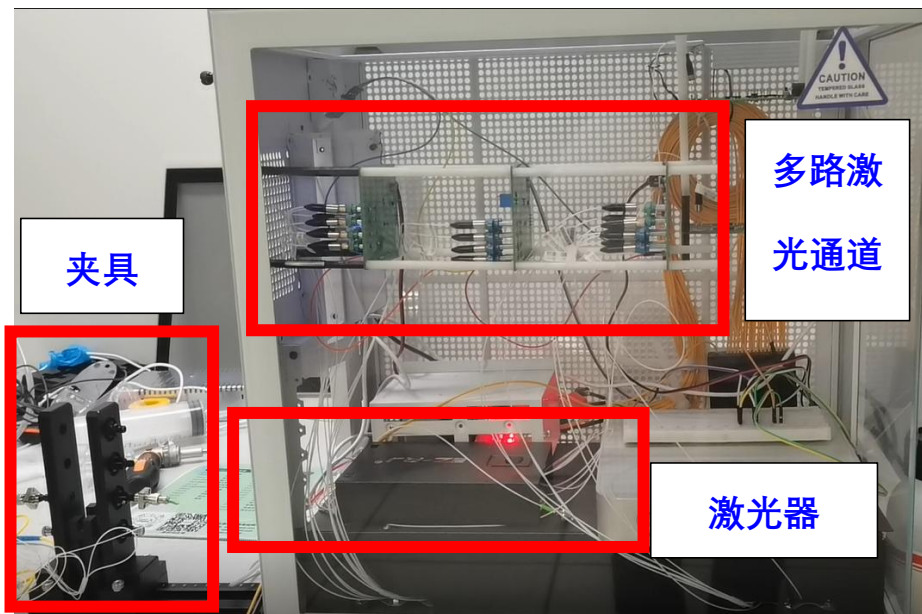


图 6

3.2 硬件系统功能展示

关于硬件系统部分，我们组成员已进行大量的数据测量，得到充分的数据集。并且基于该硬件系统，小组成员以第一作者身份发表 SCI 论文《A Fast-response Ultra-Violet Photo-transistor With a PVK QDs/ZnO Nanowire Heterostructure and Its Application in Pharmaceutical Solute Detection》。下面展示硬件系统的实现过程。

针对待测成分的微弱光谱吸收信号，设计了能够在锁定并扫描光谱波段的同时，进行高精度二次谐波信号提取的电路系统，针对多波段的激光信号，将大量淹没在背景噪声中的光信号通过差分放大模块提取有效信号，并将信号通过锁相放大电路，避开 $1/f$ 噪声和直流漂移的不利影响，通过射频选择功能，实现了一组锁相放大电路对于不同正交频率的激光信号的解调，提取对应的二次谐波信号。

通过分析相敏检波原理可知，提取二次谐波信号就需要选择参考信号频率为 $2f$ ，本系统在射频选择电路中已经实现了针对不同待测信号的选频功能与倍频功能，被测信号与参考信号分别通过对应的通道进入相敏检波电路，因为只有二次谐波中的直流分量可以通过滤波器，而含有频率分量的部分都将被低通滤波器滤

除掉，因此可以得到需要的二次谐波信号，最后提取到的二次谐波信号通过 AD 转换成数字信号输出。

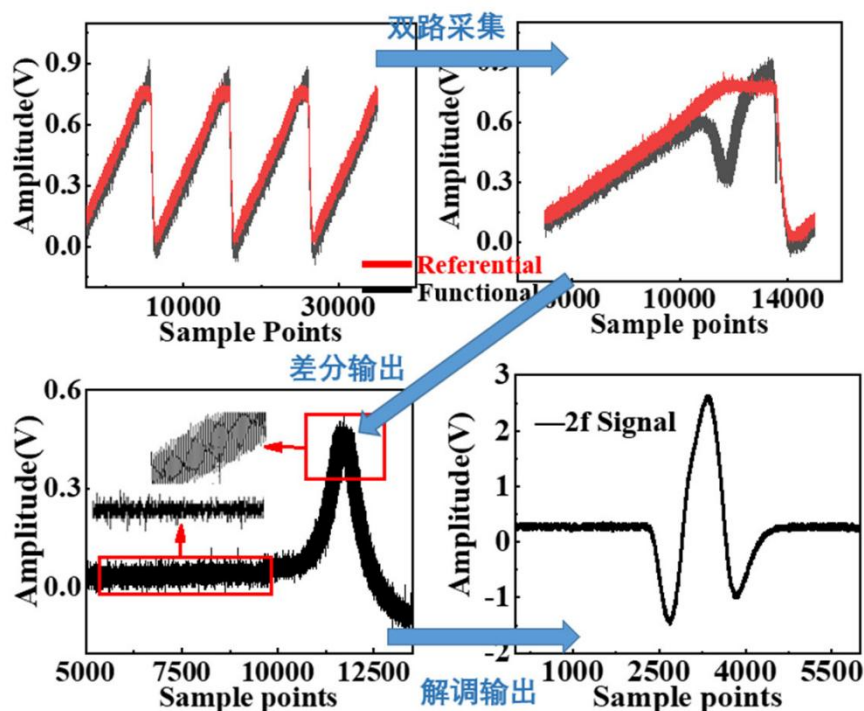


图 7 二次谐波采集过程展示

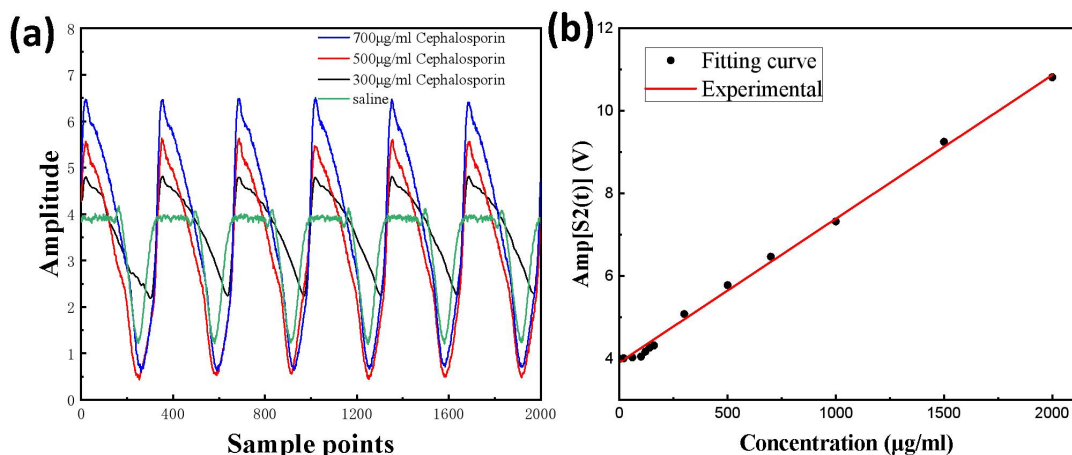


图 8 (a) 电压与待测成分浓度的关系； (b) 二次谐波信号电压幅值与待测成分浓度的线性拟合

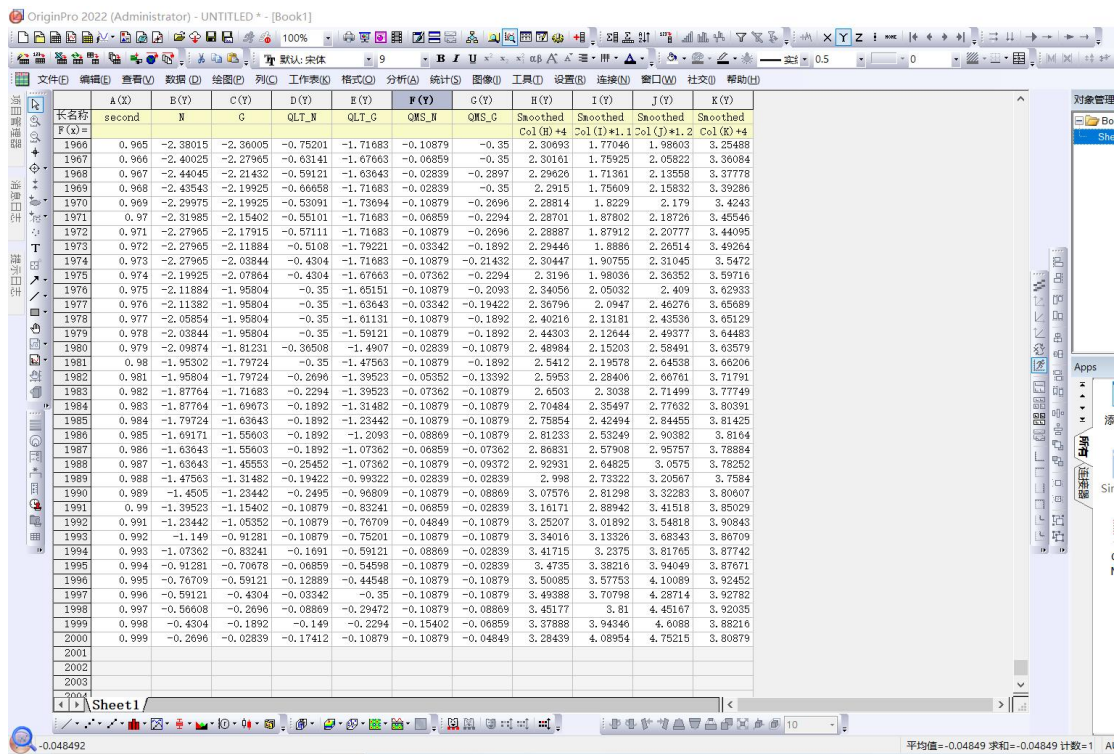
拟合结果如图 7 (b) 所示，该图是二次谐波信号与待测浓度成分的线性拟合，也是判断待测成分浓度的主要依据。

3.3 线性回归展示

由图 8 (b) 所测得拟合数据可知，每个自变量与因变量之间都满足线性关系，并且不同自变量对因变量的影响可以直接相加。因此，我们选择了线性回归模型对数据进行分析判断。具体实现代码如附录所示。

```
Starting gradient descent at b =0, w = 0, error = 0.0001
Running...
Iterations:5000: b = 0.323944, w = 1.47313, error = 112.477
-----
Process exited after 4.795 seconds with return value 0
```

图 9 极限学习机训练结果展示



长名称	A(X)	B(Y)	C(Y)	D(Y)	E(Y)	F(Y)	G(Y)	H(Y)	I(Y)	J(Y)	K(Y)
second	N	G	QLT_N	QLT_G	QMS_N	QMS_G	Smoothed Col(I)+4	Smoothed Col(I)+1.1	Smoothed Col(I)+1.2	Smoothed Col(I)+4	
F(x)=											
1966	0.965	-2.38015	-2.36005	-0.75201	-1.71683	-0.10879	-0.35	2.30693	1.77046	1.98603	3.25488
1967	0.966	-2.40025	-2.27965	-0.63141	-1.67663	-0.06859	-0.35	2.30161	1.75925	2.05822	3.36084
1968	0.967	-2.44045	-2.21432	-0.59121	-1.63643	-0.02839	-0.2897	2.29626	1.71361	2.13558	3.37778
1969	0.968	-2.43543	-2.19925	-0.66658	-1.71683	-0.02839	-0.35	2.2915	1.75609	2.15832	3.39286
1970	0.969	-2.29975	-2.19925	-0.53091	-1.73694	-0.10879	-0.2696	2.28814	1.8229	2.179	3.4243
1971	0.97	-2.31985	-2.15402	-0.58101	-1.71683	-0.06859	-0.2294	2.28701	1.87802	2.18726	3.45546
1972	0.971	-2.27965	-2.17915	-0.57111	-1.71683	-0.10879	-0.2696	2.28857	1.87912	2.20777	3.44095
1973	0.972	-2.27965	-2.11884	-0.5108	-1.79221	-0.03342	-0.1892	2.29446	1.8886	2.26514	3.49264
1974	0.973	-2.27965	-2.03844	-0.4304	-1.71683	-0.10879	-0.21432	2.30447	1.90755	2.31045	3.5472
1975	0.974	-2.19925	-2.07864	-0.4304	-1.67663	-0.07362	-0.2294	2.3196	1.98036	2.36352	3.59716
1976	0.975	-2.11884	-1.95804	-0.35	-1.65151	-0.10879	-0.2093	2.34056	2.05032	2.409	3.62933
1977	0.976	-2.11382	-1.95804	-0.35	-1.63643	-0.03342	-0.19422	2.36796	2.0947	2.46276	3.65689
1978	0.977	-2.05854	-1.95804	-0.35	-1.61131	-0.10879	-0.1892	2.40216	2.13181	2.43536	3.65129
1979	0.978	-2.03844	-1.95804	-0.35	-1.59121	-0.10879	-0.1892	2.44303	2.12644	2.49377	3.64483
1980	0.979	-2.09874	-1.81231	-0.30508	-1.4907	-0.02839	-0.10879	2.48984	2.15203	2.58491	3.63579
1981	0.98	-1.95302	-1.79724	-0.35	-1.47563	-0.10879	-0.1892	2.5412	2.19578	2.64538	3.66206
1982	0.981	-1.95804	-1.79724	-0.2696	-1.39523	-0.05352	-0.13392	2.5953	2.28406	2.66761	3.71791
1983	0.982	-1.87764	-1.71683	-0.2294	-1.39523	-0.07362	-0.10879	2.6503	2.3038	2.71499	3.77749
1984	0.983	-1.87764	-1.69073	-0.1892	-1.31482	-0.10879	-0.10879	2.70454	2.35497	2.77632	3.80391
1985	0.984	-1.79724	-1.63643	-0.1892	-1.23442	-0.10879	-0.10879	2.75854	2.42494	2.84455	3.81425
1986	0.985	-1.69171	-1.55603	-0.1892	-1.2093	-0.08869	-0.10879	2.81233	2.53249	2.90382	3.8164
1987	0.986	-1.63643	-1.55603	-0.1892	-1.07362	-0.06859	-0.07362	2.86831	2.57908	2.95757	3.78884
1988	0.987	-1.63643	-1.45553	-0.25452	-1.07362	-0.10879	-0.09372	2.92931	2.64825	3.0575	3.78252
1989	0.988	-1.47563	-1.31482	-0.19422	-0.99322	-0.02839	-0.02839	2.998	2.73322	3.20567	3.7584
1990	0.989	-1.4505	-1.23442	-0.2495	-0.96809	-0.10879	-0.08869	3.07576	2.81298	3.32283	3.80607
1991	0.99	-1.39523	-1.15402	-0.10879	-0.83241	-0.06859	-0.02839	3.16171	2.88942	3.41518	3.85029
1992	0.991	-1.23442	-1.05352	-0.10879	-0.76709	-0.04849	-0.10879	3.25207	3.01892	3.54818	3.90843
1993	0.992	-1.1149	-0.91281	-0.10879	-0.75201	-0.10879	-0.10879	3.34016	3.13326	3.68343	3.96709
1994	0.993	-1.07362	-0.83241	-0.1691	-0.59121	-0.08869	-0.02839	3.41715	3.2375	3.81765	3.87742
1995	0.994	-0.81281	-0.76709	-0.06859	-0.54598	-0.10879	-0.02839	3.4735	3.38216	3.94049	3.87671
1996	0.995	-0.76709	-0.59121	-0.12889	-0.44548	-0.10879	-0.10879	3.50085	3.57753	4.10089	3.92452
1997	0.996	-0.59121	-0.4304	-0.03342	-0.35	-0.10879	-0.10879	3.49388	3.70798	4.28714	3.92782
1998	0.997	-0.56608	-0.2696	-0.08869	-0.29472	-0.10879	-0.08869	3.45177	3.81	4.45167	3.92035
1999	0.998	-0.4304	-0.1892	-0.149	-0.2294	-0.15402	-0.06859	3.37888	3.94340	4.6088	3.88216
2000	0.999	-0.2696	-0.02839	-0.17412	-0.10879	-0.10879	-0.04849	3.28439	4.08954	4.75215	3.80879
2001											
2002											
2003											
2004											

图 10 数据集展示

如图 10 所示是我们测试的数据集，每个种类的物质有两千的数据，确保了样本容量的可靠性。

3.4 LCD 显示结果展示

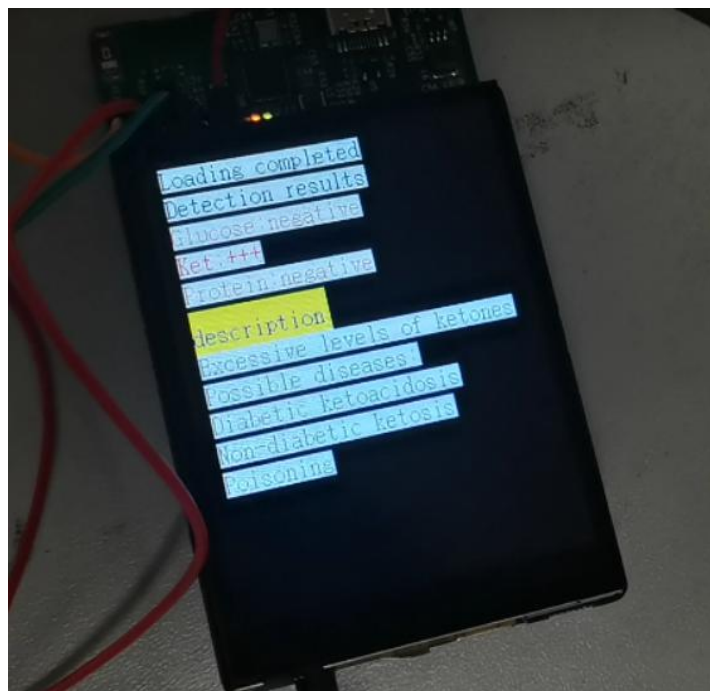


图 10Ket 超标时显示界面

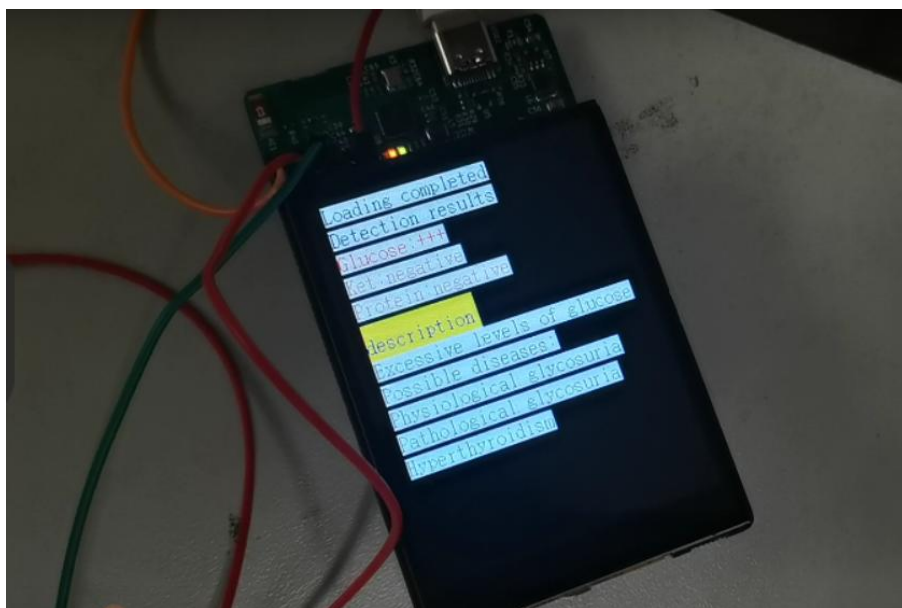


图 11 葡萄糖超标时显示界面

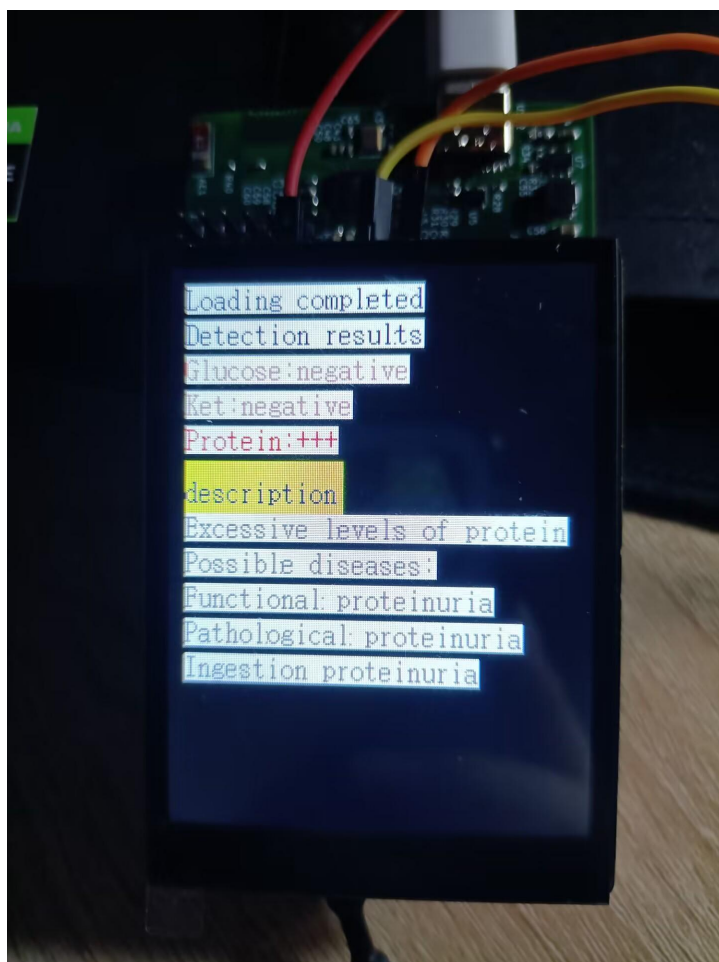


图 12 蛋白质超标时显示界面

3.5 onenet 结果显示

Results
3:Detection results: Glucose:negative; Ket:+++; Protein:negative; 2023-07-17 22:45
2:Detection results: Glucose:+++; Ket:negative; Protein:negative; 2023-07-17 22:45
1:Detection results: Glucose:negative; Ket:negative; Protein:+++; 2023-07-17 22:44
failed 2023-07-17 22:43
failed 2023-07-17 22:42
failed 2023-07-17 22:42
2:Detection results: Glucose:+++; Ket:negative; Protein:negative; 2023-07-17 22:42
1:Detection results: Glucose:negative; Ket:+++; Protein:negative; 2023-07-17 22:40
1:Detection results: Glucose:negative; Ket:+++; Protein:negative; 2023-07-17 22:35
1:Detection results: Glucose:negative; Ket:+++; Protein:negative; 2023-07-17 22:34
1:Detection results: Glucose:negative; Ket:+++; Protein:negative; 2023-07-17 22:28

图 13 手机 app 显示界面
每次设备重启时都会从 1 开始计数，当所测电压过小时（空载），显示 failed。

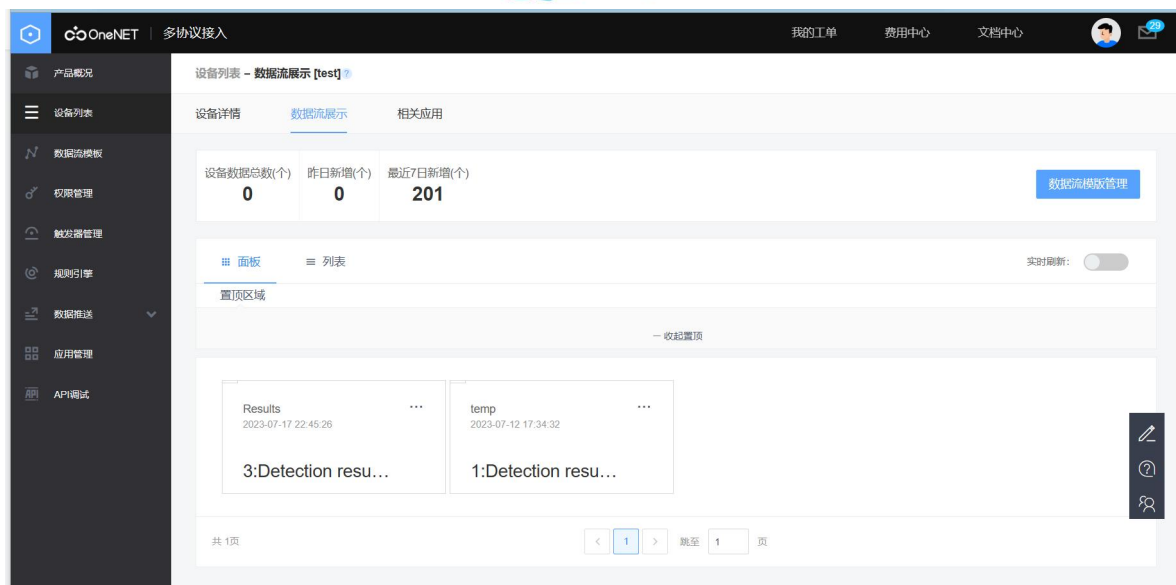


图 14 设备云端显示界面

第四部分 总结

2.4 可扩展之处

- (1)、待测指标可以增添，目前由于时间原因只测量了三种成分的数据集，后续可以再添加更多种类成分的数据集。
- (2)、LCD 显示可以优化，可以移植 LVGL，美化界面，增强人机交互功能。
- (3)、在机器学习方面，可以优化算法，引入更精确的方法，如 SVM 等。
- (4)、构建独立的、更加可视化的云平台。

2.5 心得体会

来自国家卫健委的数据显示，预测到 2040 年，慢性肾脏病将成为全球第五位的致死病因。在中国，慢性肾脏病的患病率高达 10.8%，患者人数超过 1 亿，需要接受肾脏替代治疗的终末期肾病患者有 150 多万，并以每年新增 12 万~15 万的趋势持续上升。

尿常规是医学检验“三大常规”项目之一，不少肾脏病变早期就可以出现蛋白尿或者尿沉渣中有形成分。对于某些全身性病变以及身体其他脏器影响尿液改变的疾病如糖尿病、血液病、肝胆疾患、流行性出血热等的诊断，也有很重要的参考价值。同时，尿液的化验检查还可以反映一些疾病的治疗效果及预后。通过此项检查可以判断相应的病征。

但是，尿常规对于一些特殊人群并不是足够便捷。如果因为行动不便等原因不及时做尿常规检查，可能会导致错过肾脏疾病的最佳治疗时间。

对于具体项目的实施，我深深感到，在工程开始实施之前，要做好方案论证、可行性研究及成果价值的评估，这是非常重要且不可简单下定论的部分，当然也要依靠工程实践和经验的积累才有能力做出准确的判断。

国内有意在 RISC-V 架构上发力，为服务国家需求、紧跟技术发展的方向，面对形形色色的 SOC 芯片层出不穷，掌握一种快速通用的开发方法已经成为新

技术人才的刚需，培养阅读手册的能力和基于 SDK 的开发逐渐成为市场的主流。在基于博流 SDK 开发的过程中，培养的不仅仅是博流芯片开发的能力和嵌入式开发思维，更重要的是，拿到其他新芯片也能快速上手编程开发，从而能够紧跟新技术潮流，不再局限于某一款芯片或某一种架构。

开源给技术发展和学习注入活力。开源给开发带来了许多便利，极大缩短了开发周期，会用开源，并且愿意贡献开源，构建健康的开源生态，我们每一位开发者都应当为之共同努力。

第五部分 参考文献

- [1] Li, J.; Guo, Q.; Tao, Y.; Li, D.; Yang, Y.; Zhou, D.; Pan, J.; Liu, X.; Tao, Z. A Fast-Response Ultraviolet Phototransistor with a PVKQDs/ZnO Nanowire Heterostructure and Its Application in Pharmaceutical Solute Detection. *Nanomaterials* 2023, 13, 1364. <https://doi.org/10.3390/nano13081364>
- [2] 席浩. 轻量级人体检测与行为识别算法的研究 [D]. 电子科技大学, 2021. DOI:10.27005/d.cnki.gdzku.2021.000595.
- [3] BL MCU SDK 开发指南, https://dev.bouffalolab.com/media/doc/sdk/bl_mcu_sdk_zh/index.html.
- [4] BL602 编程指南, <https://dev.bouffalolab.com/media/doc/bliotsdk/html/index.html>.
- [5] BL602 AT 指令集与使用示例, https://dev.bouffalolab.com/media/doc/602/at_command/html/index.html.
- [6] TensorFlow Lite 指南, <https://tensorflow.google.cn/lite/guide>.
- [7] BL602/604 数据手册, <https://dev.bouffalolab.com/media/doc/602/open/datasheet/zh/html/index.html>.
- [8] BL602/604 参考手册, https://dev.bouffalolab.com/media/doc/602/open/reference_manual/zh/html/index.html.
- [9] BL702/704/706 开源数据手册, <https://dev.bouffalolab.com/media/doc/702/open/datasheet/zh/html/index.html>.
- [10] BL702/704/706 参考手册, https://dev.bouffalolab.com/media/doc/702/open/reference_manual/zh/html/index.html.

第六部分 附录

线性回归算法代码

```
#include <iostream>
#include <fstream>
#include <cmath>
using namespace std;
```

```
const int SAMPLE = 27;
const int PARAMETER = 12;
```



```
double hypoVal(double para[], double fea[], int count);
double costVal(double para[], double lab[], int amount, double allX[][PARAMETER
+ 1]);
```

```
int main()
{
    //文本操作，读入数据
    fstream infile;
    infile.open("data.txt");
    if (!infile)
    {
        cout << "can't open file!" << endl;
        return -1;
    }
    //二维数组储存数据
    double X[SAMPLE + 1][PARAMETER + 1];    //features matrix
    for (int i = 0; i < SAMPLE + 1; i++)
    {
        for (int j = 0; j < PARAMETER + 1; j++)
        {
            infile >> X[i][j];
        }
    }
    infile.close();
    //验证数据是否写入
    for (int i = 0; i < SAMPLE + 1; i++)
    {
        for (int j = 0; j < PARAMETER + 1; j++)
            cout << X[i][j] << " ";
        cout << endl;
    }
    //特征缩放

    //留一法，得到训练集 X[0]~X[26]和测试集 X[27]
    double y[SAMPLE] = { 0 };    //labels vector
    double theta[PARAMETER] = { 0 };    //parameters vector
    double a = 0.0001;    //set learning rate as 0.0001
    int cnt = 0;    //to count the times of loop
    for (int i = 0; i < SAMPLE; i++)
    {
        y[i] = X[i][PARAMETER];
    }
    double h = hypoVal(theta, X[0], PARAMETER); //hypothesis function
```



```
double cost = costVal(theta, y, SAMPLE, X); //cost function

//梯度下降法求 theta
double temp[PARAMETER] = { 0 };          //used for simultaneously updating
theta parameters
double der[PARAMETER] = { 0 }; //the derivative term
double tempCost = 0;                //break the loop when tempCost - cost is small
enough
do {
    tempCost = cost;
    double sum = 0;
    for (int j = 0; j < PARAMETER; j++)
    {
        for (int i = 0; i < SAMPLE; i++)
        {
            sum += (hypoVal(theta, X[i], PARAMETER) - y[i]) * X[i][j];
        }
        der[j] = (1.0 / double (PARAMETER)) * sum;
        temp[j] = theta[j] - a * der[j];
        sum = 0;
    }
    cout << "now the theta parameters are: ";
    for (int i = 0; i < PARAMETER; i++)
    {
        theta[i] = temp[i];
        cout << theta[i] << " ";
    }
    cout << endl;
    cost = costVal(theta, y, SAMPLE, X);          //new cost value
    cnt++;
} while (tempCost - cost > 0.00001);

//测试集进行测试
cout << "共进行" << cnt << "次梯度下降法" << endl;
h = hypoVal(theta, X[SAMPLE - 1], PARAMETER);
cout << "对测试数据预测值为: " << h << ", 真实值为: " << y[SAMPLE - 1]
<< endl;
return 0;
}

double hypoVal(double para[], double fea[], int count) //the value of hypothesis
function
{
    double hy = 0;
```

```

    for (int i = 0; i < count; i++)
    {
        hy += para[i] * fea[i];
    }
    return hy;
}
double costVal(double para[], double lab[], int amount, double allX[][PARAMETER
+ 1])    //the value of cost function
{
    double sum = 0;
    for (int i = 0; i < amount; i++)
    {
        sum += pow((hypoVal(para, allX[i], PARAMETER) - lab[i]), 2);
    }
    cout << "costVal now is : " << (1.0/(2.0 * (double) amount)) * sum << endl;
    return double(1.0/(2.0 * (double) amount)) * sum;
}

```