

内蒙古科技大学

本科生毕业设计说明书

题 目：智能家居能耗监控系统控制
 中心及 APP 设计

学生姓名：袁申强

学 号：1367165110

专 业：建筑电气与智能化

班 级：建电 2013

指导教师：李琦 教授

智能家居能耗监控系统控制中心及 APP 设计

摘 要

科学技术的发展为人类文明作出卓越贡献的同时，也带为地球来了更快的能源消耗。建筑能耗已经继交通之后成为人类社会的第三大耗能产业。在传统建筑设备自动化系统内并入能耗监测系统，灵活性不强且成本高昂。嵌入式及物联网技术的蓬勃发展为建筑物能耗监测系统的构建提供新思路。

本文以嵌入式系统和物联网技术为基础，讨论了小型实用型家用能耗监测平台的设计与实现。使用树莓派单板机搭建家庭能耗控制中心，采集水表、电表、气表和室内光照度值，提供网络后台服务，开发配套的 web app，允许通过网络访问控制中心的能耗数据以及远程操作控制室内亮度以及相关阀的通断。设计的主要内容包括 webApp 后台软件设计、能耗数据采集网关服务软件设计、webapp 前端界面设计以及嵌入式 web 服务器的搭建和系统向树莓派的移植几大部分。在组网方面采用 lora 无线通讯技术实现上位机与下位机的数据通信；上位机与无线路由器相连接入家庭网络。用户通过手机浏览器访问树莓派即可进行系统登录，访问系统数据。搭建了一个智能家居能耗监测实物演示系统，经过运行测试，预期效果基本实现，系统运转良好。

关键词: 建筑能耗；嵌入式；物联网；网络服务；移动端

Design of Home Energy Monitoring & Control System and APP

Abstract

The development of science and technology had benefit a lot to human civilization, but at the same time, a faster energy consumption follows thereupon. Building energy consumption has been following the traffic after the human society to become the third largest energy-consuming industries. Bringing the energy consumption-monitoring-system into the traditional building automation system causes a poor flexibility and appears costly. The vigorous development of embedded and Internet of Things technology provides new ideas for the construction of building energy consumption monitoring system.

Based on the embedded and Internet of Things technology, this paper discusses the realization of small practical household energy consumption monitoring platform. to build a family-energy-control center on a raspberry, collecting water meters, electric meters, gas and illuming values indoors, providing network background services for users and developping a corresponding web app. People will access to the control center to get energy-consumption data and control Indoor brightness and turn on/off the valves via remote operation. The main contents of this design included the realization of webApp back end, energy-consumption data acquisition gateway, webapp front-end interface ,the construction of embedded web server and transplanting the system to raspberry. As to the networking-building, using lora wireless communication technology to achieve data communication between the host computer and the appendant devices of; the host connected to earthnet router in house. Users can login the system through the mobile browser to access the raspberry and system data. After running the test, the expected effect is basically achieved, the system works well.

Key words: building energy consumption; embedded; Internet of things; network service; mobile terminals

目 录

摘要	I
Abstract	II
第一章 绪论	1
1.1 选题背景	1
1.2 国内外相关研究现状	1
1.2.1 国外相关研究现状	1
1.2.2 国内相关研究现状	1
1.3 课题研究的内容	1
1.4 论文结构	1
第二章 XXX 系统的总体设计	2
2.1 需求分析	2
2.2 XXX 系统的总体设计方案	2
2.3 XXX 其他总体设计相关内容 1	2
2.4 XXX 其他总体设计相关内容 2	2
2.5 本章小结	2
第三章 XXXX 系统硬件设计	3
3.1 单片机最小系统设计	3
3.2 传感器选型及其单片机接口电路设计	3
3.2.1 TLC549 八位 AD 转换芯片介绍	3
3.2.2 TLC549 管脚功能及 AD 转换电路原理图	3
3.2.3 数字电压表电路原理图	4
3.3 无线通信模块选型及其接口电路设计	5
3.4 本章小结	5
第四章 XXXX 系统软件设计	6
4.1 主程序设计	6
4.2 系统初始化子程序设计	7
4.3 中断服务子程序设计	7
4.4 AD 转换子程序设计	7

4.5	关键算法	7
4.5.1	标度变换算法	7
4.5.2	更新显示缓冲区算法	8
4.6	传感器数据采集程序设计	8
4.7	无线通信协议及程序设计	8
4.8	本章小结	8
第五章	系统调试	9
5.1	本章小结	9
第六章	总结	10
第七章	模板的使用方法	11
7.1	具体使用步骤	11
7.2	编译的方法	11
7.3	Readme	11
7.4	字体调节	11
7.5	字号调节	12
7.6	列表演示	12
7.7	已加入的常用宏包	13
7.8	中英文间距问题	13
7.9	引用的问题	13
7.9.1	参考文献的引用	13
7.9.2	公式及其引用	14
7.10	图形与表格	14
7.11	关于论文中的算法和代码	16
7.11.1	算法	16
7.11.2	代码	16
7.12	FAQ	18
7.13	其他事项	18
参考文献	20
附录 A	电路原理图	21
附录 B	源代码	22

致谢 25

第一章 绪论

1.1 选题背景

介绍选题的背景，1-2 页，不超过 2 页。

1.2 国内外相关研究现状

1.2.1 国外相关研究现状

1-2.5 页，介绍国外相关研究现状，注意参考文献标注!!!

1.2.2 国内相关研究现状

1.5-2.5 页，介绍国内相关研究现状，注意参考文献标注!!!

1.3 课题研究的内容

0.5-1 页，介绍本课题重点研究的内容

1.4 论文结构

一段话，介绍论文章节安排，每章一句话。

第二章 XXX 系统的总体设计

2.1 需求分析

从用户的角度而不是设计师的角度描述 XXX 系统要实现哪些具体功能，给出主要技术参数。可给出一张总体的需求图。

2.2 XXX 系统的总体设计方案

给出 XXX 系统的总体设计方案，画设计方案方框图。比如选择了 STM32 单片机作为主控制器，围绕它扩展哪些外围接口等。包括但不限于用到的相关软硬件技术、理论知识、关键算法、开发工具等等

2.3 XXX 其他总体设计相关内容 1

2.4 XXX 其他总体设计相关内容 2

2.5 本章小结

一段话小结本章内容。

第三章 XXXX 系统硬件设计

3.1 单片机最小系统设计

3.2 传感器选型及其单片机接口电路设计

3.2.1 TLC549 八位 AD 转换芯片介绍

AD 转换芯片的作用^[1]是将模拟信号转换为数字信号^[2],TLC549 是 8 位串行 A/D 转换器芯片,可与通用微处理器、控制器通过 CLK、CS、DATAOUT 三条口线进行串行接口。具有 4MHz 片内系统时钟和软、硬件控制电路,转换时间最长 17 μ s, TLC549 为 40000 次/s。总失调误差最大为 ± 0.5 LSB, 典型功耗值为 6mW。采用差分参考电压高阻输入, 抗干扰, 可按比例量程校准转换范围, VREF-接地, VREF+ - VREF ≥ 1 V, 可用于较小信号。

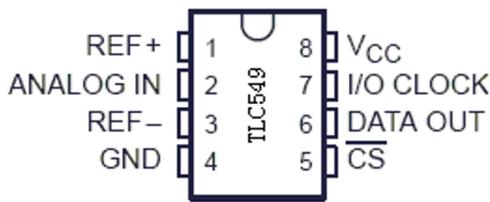


图 3.1 TLC549 八位 AD 转换芯片

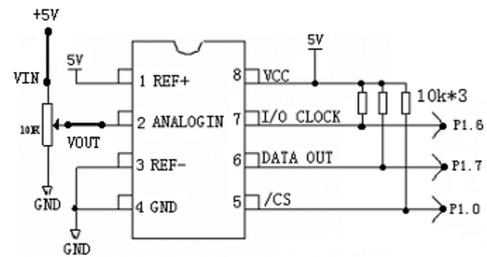


图 3.2 TLC549AD 转换电路原理图

3.2.2 TLC549 管脚功能及 AD 转换电路原理图

表 3.1 TCL549 芯片管脚定义

管脚名称	描述
REF+	正基准电压输入, $2.5V \leq REF+ \leq V_{cc} + 0.1$ 。
REF -	负基准电压输入端, $-0.1V \leq REF- \leq 2.5V$ 。
VCC	系统电源 $3V \leq V_{cc} \leq 6V$ 。
GND	接地端。
/CS	芯片选择输入端。
DATA OUT	转换结果数据串行输出端, 高位在前, 低位在后。
ANALOG IN	模拟信号输入端, $0 \leq ANALOGIN \leq V_{cc}$ 。
I/O CLOCK	输入/输出时钟输入端, 同于同步芯片的输入输出操作。

TLC549 的管脚功能定义如表 3.1 所示，其管脚分布如图 3.1 所示。

单片机扩展 TLC549AD 转换电路原理图如图 3.2 所示，单片机 I/O 口分别与 I/O CLOCK、DATA OUT 及 /CS 相连，模拟量收入端接 ANALOG IN，REF+、REF- 分别接 +5V 和 GND。

3.2.3 数字电压表电路原理图

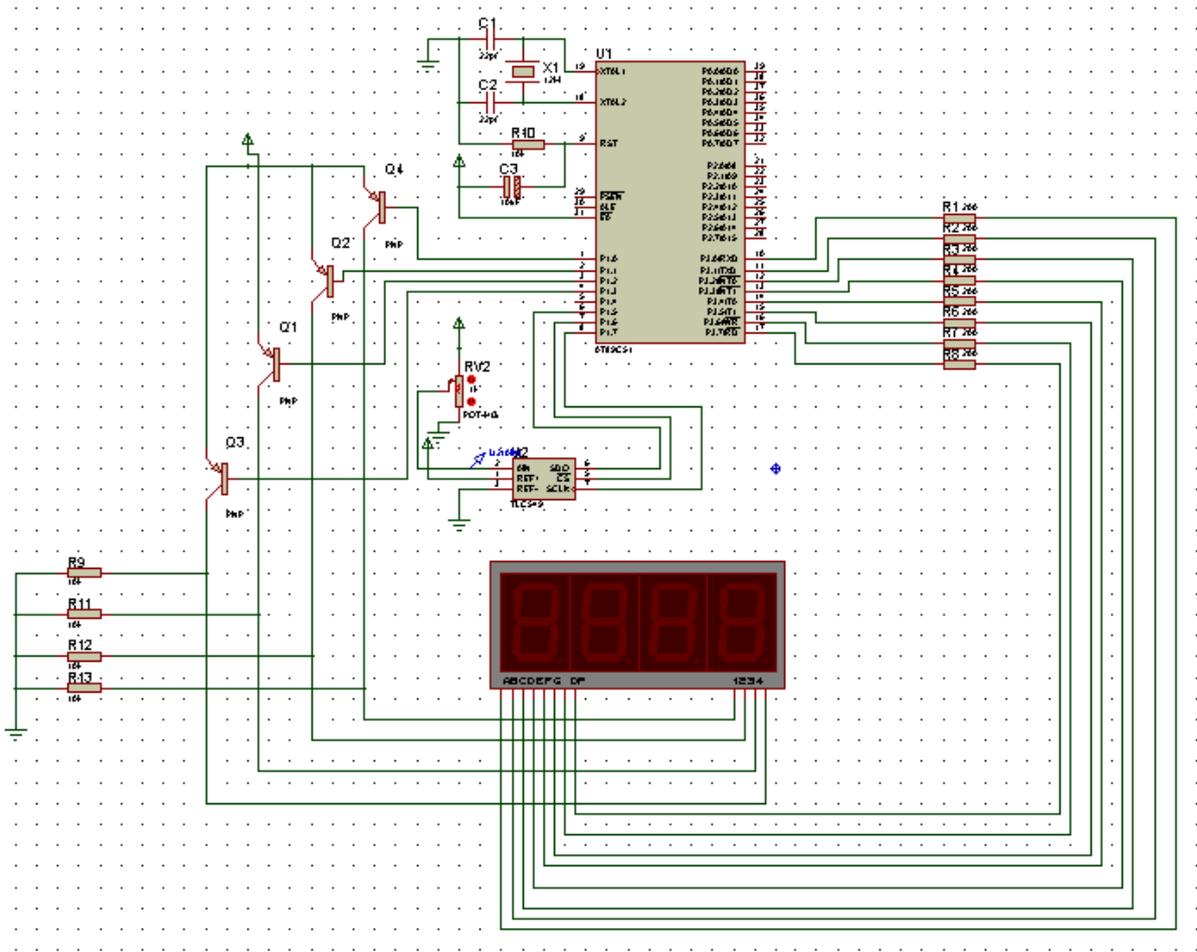


图 3.3 数字电压表电路原理图

数字电压表的电路原理图如图 3.3 所示，包括由电源电路、时钟电路及复位电路组成的单片机最小系统电路，AD 转换接口电路及 4 位数码管驱动电路。P1.5-P1.7 分别与 TLC549 的 CLK、/CS 及 DOUT 相连，4 位数码管驱动电路采用动态刷新工作模式，P1.0-P1.3 输出 LED 数码管驱动电路的字位码，P3 口输出字型码。

3.3 无线通信模块选型及其接口电路设计

3.4 本章小结

第四章 XXXX 系统软件设计

4.1 主程序设计

主程序流程图如图 4.1 所示, 首先调用系统初始化子程序, 这里主要是对定时器 T0 初始化, T0 中断产生两个时标, 1 个 2ms 的时标 flag 用于刷新 4 位 LED 数码管, 1 个 80ms 时标 flag1 用于更新电压显示值。80ms 时标到时间, 读取当前 AD 值, 进行标度变换并更新数据显示缓冲区。2ms 时间到, 通过 P3 口和 P1 口分别输出字型码和字位码, 实现 4 位 LED 数码管的动态刷新。

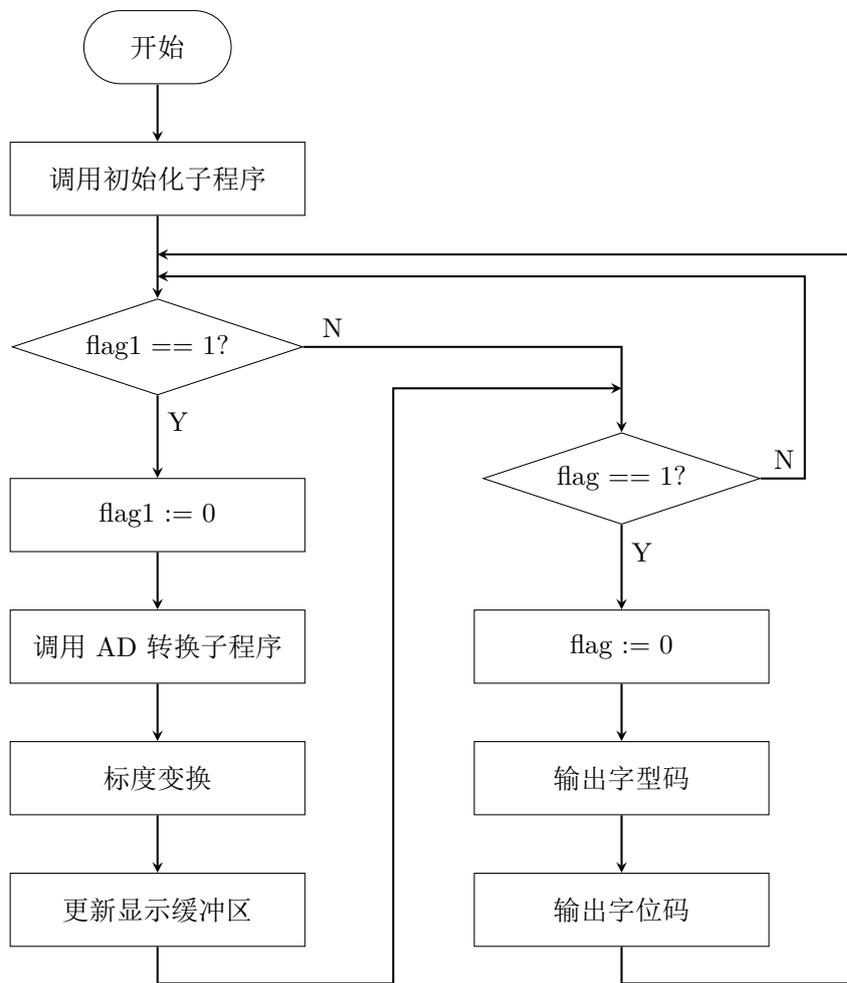


图 4.1 主程序流程图

4.2 系统初始化子程序设计

4.3 中断服务子程序设计

4.4 AD 转换子程序设计

当/CS 有效后，TLC549 输出最高位 D7，之后 TLC549_CLK 每输出 1 个下降沿信号，TLC549 依次输出 D6-D0，经过 7 个下降沿信号和左移操作获得 AD 值。

AD 转换子程序代码如下所示：

```
1 uint8_t getAdValue(){
2     uint8_t i;
3     TLC549_CS = 0;
4     TLC549_DOUT = 1;
5     TLC549_CLK = 0;
6     for(i=0;i<8;i++){
7         AdValue = AdValue<<1;
8         TLC549_CLK = 1;
9         _nop_();
10        AIN = TLC549_DOUT;
11        TLC549_CLK = 0;
12        _nop_();
13    }
14    TLC549_CS = 1;
15    return AdValue;
16 }
```

4.5 关键算法

4.5.1 标度变换算法

标度变换就是将 AD 转换芯片返回的数字量转换为对应物理量的过程。

Y_{max} : 物理量的最大值, Y_{min} : 物理量的最小值, X_{max} :AD 的最大值, X_{min} :AD 的最小值, z 为当前 AD 返回值。

则: $\frac{Y_{max}-Y_{min}}{X_{max}-X_{min}}$ 为 1 格 AD 值对应的物理量大小, 在本例中为 $\frac{5.0-0.0}{255.0-0.0}$, 加小数点的原因是将其设为浮点类型, 否则计算机会取整, 丢失精度。

因此，标度变换后的物理量 = $z \times \frac{Y_{max}-Y_{min}}{X_{max}-X_{min}}$ 。

在程序实现中, 需要强制转换为 float 类型, 为了便于传输和显示, 将其放大 1000 倍取整 (保留三位小数)。

标度变换核心代码如下所示:

```
1 if(adFlag){
2     adFlag = 0;
3     fAdValue = (float)(ucAdValue * (5.0/255.0));
4     uiAdValue= fAdValue * 1000;
5 }
```

4.5.2 更新显示缓冲区算法

uiAdValue 返回的是放大 1000 倍的整型值, 需要将其千位, 百位, 十位, 个位拆分开以更新显示缓冲区。

通过除以 10 取余获得个位, 通过除以 10 取整经过 4 次循环分别取出千位, 百位, 十位, 个位并更新到缓冲区。

更新显示缓冲区核心代码如下所示:

```
1 for(j = 0; j < 4; j++){
2     dispBuf[3-j] = uiAdValue%10;
3     uiAdValue /= 10;
4 }
```

4.6 传感器数据采集程序设计

4.7 无线通信协议及程序设计

4.8 本章小结

第五章 系统调试

本章写出系统调试的过程，遇到的问题及解决的办法，给出演示系统的图片。

5.1 本章小结

第六章 总结

1 页以内

第七章 模板的使用方法

7.1 具体使用步骤

1. 进入 includefile 文件夹, 打开 midmatter.tex, backmatter.tex 这两个文档, 分别填写中文摘要、英文摘要和致谢。
2. 打开主文档 MasterTemplate.tex, 填写题目、作者等等信息, 书写正文。
3. 使用 XeLaTeX 编译. 具体见 7.2。

7.2 编译的方法

默认使用 XeLaTeX 编译, 直接生成 pdf 文件.

若另存为新文档, 请确保文档保存类型为 :UTF-8. 当然目前很多编辑器默认文字编码为 UTF-8. WinEdt 9.0 之后的版本都是默认保存为 UTF-8 的.

7.3 Readme

模板文件的结构, 如下表所示:

IMUSTbachelor.tex		主文档. 在其中填写正文.
includefile 文件夹	midmatter.tex	中文摘要, 英文摘要. (自行填写)
	backmatter.tex	致谢. (自行填写)
figures 文件夹		存放图片文件.
BIBbase 文件夹		供 BibTeX 做参考文献时选用.
IMUSTBachelor.cls		定义文档格式的 class file. 不可删除.

无需也不要改变、移动上述文档的位置.

7.4 字体调节

\songti 宋体
 \heiti 黑体
 \fangsong 仿宋
 \kaishu 楷书

7.5 字号调节

字号命令: \zihao

\zihao{0}	初号字 English
\zihao{-0}	小初号 English
\zihao{1}	一号字 English
\zihao{-1}	小一号 English
\zihao{2}	二号字 English
\zihao{-2}	小二号 English
\zihao{3}	三号字 English
\zihao{-3}	小三号 English
\zihao{4}	四号字 English
\zihao{-4}	小四号 English
\zihao{5}	五号字 English
\zihao{-5}	小五号 English
\zihao{6}	六号字 English
\zihao{-6}	小六号 English
\zihao{7}	七号字 English
\zihao{8}	八号字 English

7.6 列表演示

第一种：数字.

1. 列表项 1。
2. 列表项 2。
3. 列表项 3。

第二种：(数字)

- (1) 列表项 1。
- (2) 列表项 2。
- (3) 列表项 3。

第三种：(罗马顺序)

- (i) 列表项 1。
- (ii) 列表项 2。
- (iii) 列表项 3。

第三种：(英文字母顺序)

- (a) 列表项 1。
- (b) 列表项 2。
- (c) 列表项 3。

7.7 已加入的常用宏包

cite 参考文献引用, 得到形如 [3-7] 的样式.

color,xcolor 支持彩色.

enumerate 方便自由选择 enumerate 环境的编号方式. 比如

`\begin{enumerate}[(a)]` 得到形如 (a), (b), (c) 的编号.

`\begin{enumerate}[i]` 得到形如 i), ii), iii) 的编号.

另外要说明的是, `itemize`, `enumerate`, `description` 这三种 list 环境, 已经调节了其间距和缩进, 以符合中文书写的习惯.

7.8 中英文间距问题

自动加入间距. 不再需要在公式、英文前后加字符“~”或空格.

7.9 引用的问题

7.9.1 参考文献的引用

参考文献的引用, 用命令 `\cite{ }`. 大括号内要填入的字串, 是自命名的文献条目名.

比如, 通常我们会说:

关于此问题, 请参见文献^[3]. 作者某某还提到了某某概念^[4].

上文使用的源文件为:

关于此问题, 请参见文献 `\cite{r6}`.

另外,要得到形如^[4-8]的参考文献连续引用,需要用到 cite 宏包(模板已经加入),在正文中使用 `\cite{r1,r3,r4,r5}` 的引用形式即可.

引用效果^[4,6-8]。

引用效果^[3,6-8]。

引用效果^[9,10]。

引用效果^[11]。^[12]^[13]

7.9.2 公式及其引用

单独的公式示例:

$$\sum_{i=1}^n a_i = 0 \quad (7.1)$$

$$f(x_1, x_2, \dots, x_n) = x_1^2 + x_2^2 + \dots + x_n^2 \quad (7.2)$$

$$\sum_{k=0}^{j-1} \hat{\gamma}_{kj} z_k \quad (7.3)$$

式 (7.2) 的引用示例!

式 7.1 的引用示例!

7.10 图形与表格

支持 eps, pdf, png,jpg 这几种常见图形格式.

再次澄清一个误会: \LaTeX 支持的图形格式绝非 eps 这一种. 无需特意把图片转化为 eps 格式.

用形如 `\includegraphics[width=12cm]{Daisy.jpg}` 的命令可以纳入图片.

如图 7.2 是一个纳入 jpg 图片的例子.

一个 4 个子图排列的例子, 如图 7.1 所示。

图 7.1(a) 为 `batch_size = 1(epoch=16)` 的训练过程的 loss 曲线。

图 7.1(b) 为 `batch_size = 4(epoch=16)` 的训练过程的 loss 曲线。

图7.1(c) 为 `batch_size = 8(epoch=16)` 的训练过程的 loss 曲线。

图 7.1(d) 为 `batch_size = 16(epoch=16)` 的训练过程的 loss 曲线。

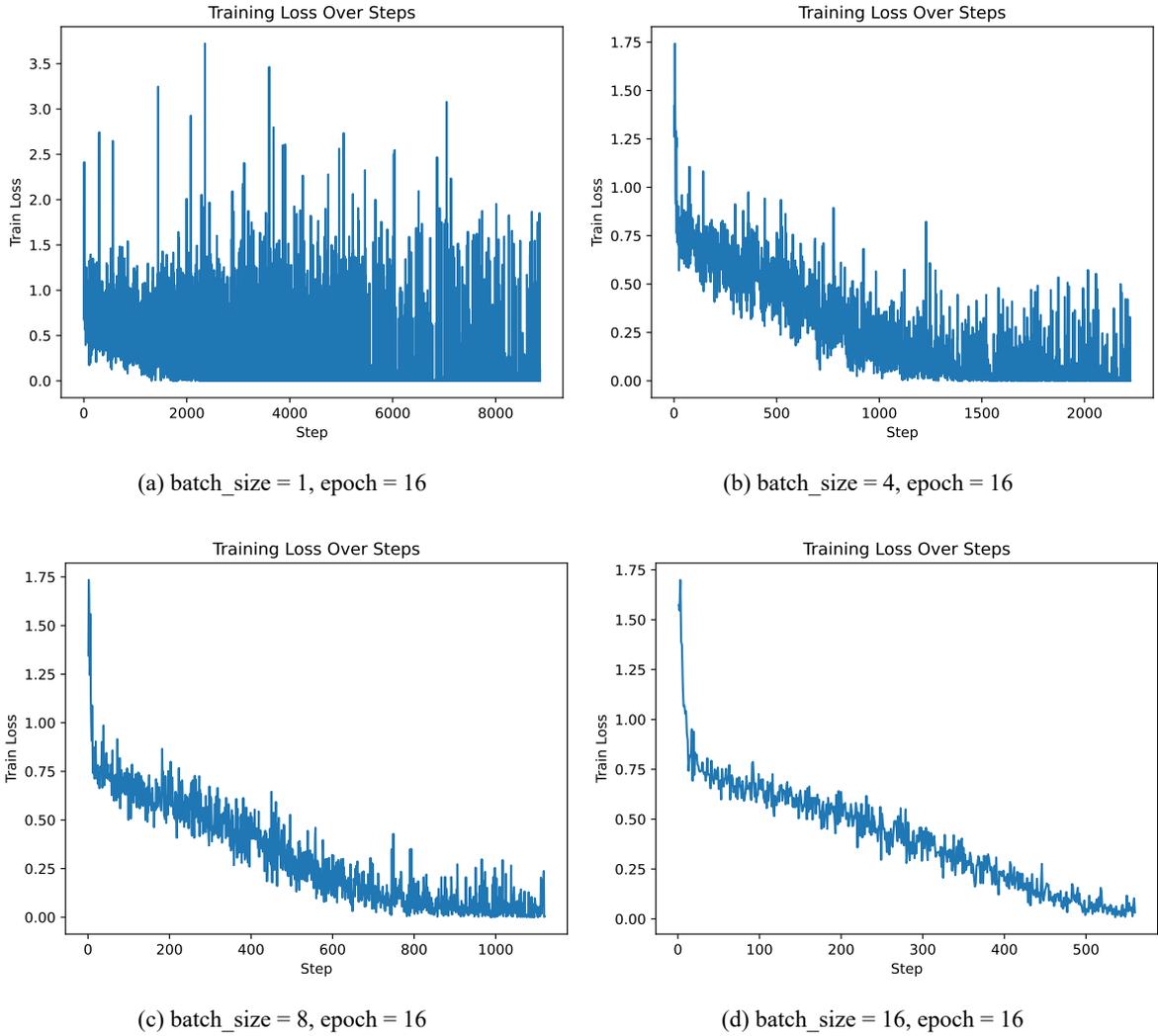


图 7.1 眼科单选题 LoRA 微调训练过程的 loss 变化曲线 (训练集 2213 条)

表格问题, 建议使用“三线表”, 如表 7.1.

表 7.1 一般三线表

123	4	5	123	4	5	123	4	5	123	4	5
67	890	13	123	4	5	123	4	5	123	4	5
67	890	13	123	4	5	123	4	5	123	4	5
67	890	13	123	4	5	123	4	5	123	4	5

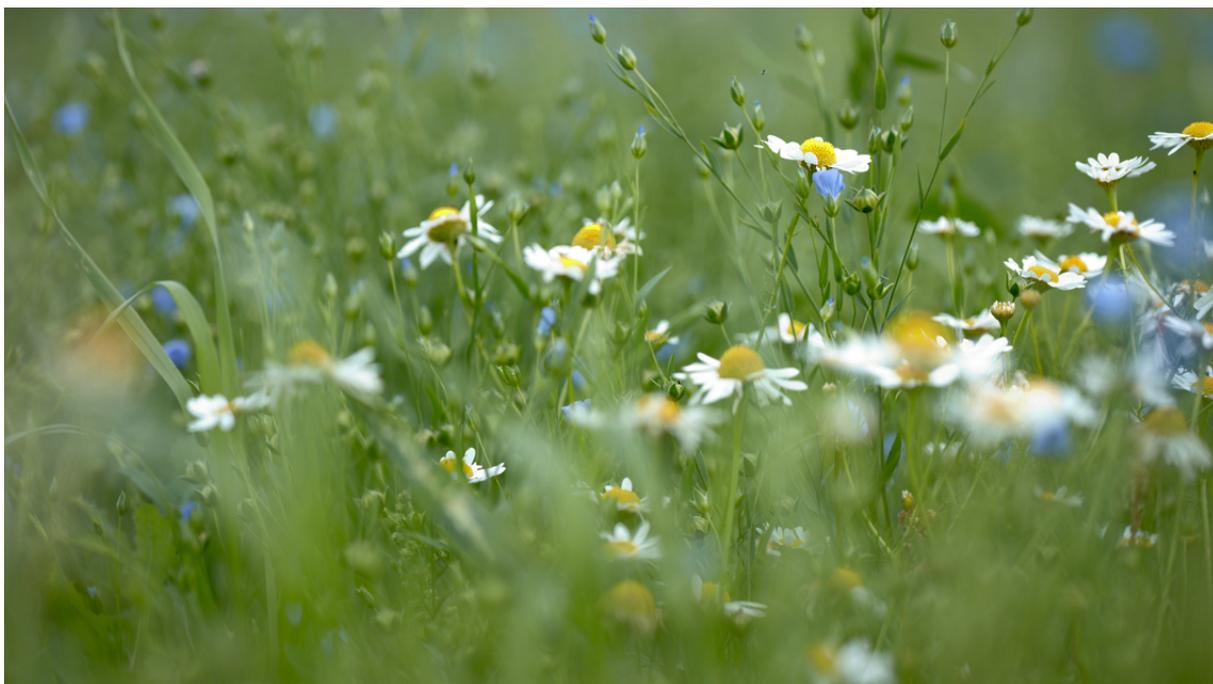


图 7.2 一个彩色 jpg 图片的例子

7.11 关于论文中的算法和代码

7.11.1 算法

描述算法采用“algorithm”环境实现，用来表达算法的伪代码。

k 均值算法是典型的原型聚类算法，它将聚类问题转化为最优化问题。具体做法是先找到 k 个聚类中心，并将所有样本分配给最近的聚类中心，分配的原则是让所有样本到其聚类中心的距离平方和最小。显然，距离平方和越小意味着每个聚类内样本的相似度越高。但是这个优化问题没有办法精确求解，因而只能搜索近似解。k 均值算法就是利用贪心策略，通过迭代优化来近似求解最小平方和的算法。

7.11.2 代码

正文部分不宜放入大篇幅代码，确需通过代码阐明内容的，建议每段代码不超过 1 页，大段代码可以放在附录中。

代码采用“lstlisting”实现。注意代码的排版规范和缩进，由于“lstlisting”环境不能对代码自动排版，需要将排好版的代码填入“lstlisting”中，可以采用 VSCODE 的代码格式化插件实现代码的排版。

C 语言代码的示例如下：

算法 1: k -均值算法

输入: 样本集 $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$ 、聚类簇数 k .

输出: 簇划分 $\mathcal{C} = \{C_1, C_2, \dots, C_k\}$.

```

1 从  $D$  中随机选择  $k$  个样本作为初始均值向量  $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_k\}$ ;
2 repeat
3   令  $C_i = \emptyset$  ( $1 \leq i \leq k$ );
4   for  $j = 1, 2, \dots, m$  do
5     计算样本  $\mathbf{x}_j$  与各均值向量  $\boldsymbol{\mu}_i$  ( $1 \leq i \leq k$ ) 的距离:  $d_{ji} = \|\mathbf{x}_j - \boldsymbol{\mu}_i\|_2$ ;
6     根据距离最近的均值向量确定  $\mathbf{x}_j$  的簇标记:  $\lambda_j = \arg \min_{i \in \{1, 2, \dots, k\}} d_{ji}$ ;
7     将样本  $\mathbf{x}_j$  划入相应的簇:  $C_{\lambda_j} = C_{\lambda_j} \cup \{\mathbf{x}_j\}$ ;
8   end
9   for  $i = 1, 2, \dots, k$  do
10    计算新均值向量:  $\boldsymbol{\mu}'_i = \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{x}$ ;
11    if  $\boldsymbol{\mu}'_i \neq \boldsymbol{\mu}_i$  then
12      将当前均值向量  $\boldsymbol{\mu}_i$  更新为  $\boldsymbol{\mu}'_i$ ;
13    else
14      保持当前均值向量不变;
15    end
16  end
17 until 当前均值向量不再更新;

```

```

1
2 for (j = 0; j < 4; j++) {
3   dispBuf[3 - j] = uiAdValue % 10;
4   uiAdValue /= 10;
5 }

```

python 语言代码的示例如下:

```

1 #求解梯度
2 def gradient(theta, X, y):
3     m = y.size
4     h = sigmoid(X.dot(theta.reshape(-1,1)))
5     grad = (1.0/m)*X.T.dot(h-y)
6     return(grad.flatten())

```

```

7 initial_theta = np.zeros(X.shape[1])
8 cost = costFunction(initial_theta, X, y)
9 grad = gradient(initial_theta, X, y)
10 res = minimize(costFunction, initial_theta, args=(X,y), jac=gradient,
    options={'maxiter':400})
11
12 def predict(theta, X, threshold=0.5):
13     p = sigmoid(X.dot(theta.T)) >= threshold
14     return(p.astype('int'))
15
16 sigmoid(np.array([1, 45, 85]).dot(res.x.T))

```

7.12 FAQ

(1) 为什么我的图表都跑到最后面去了?

LaTeX 的排版算法会根据文字和图表占用的空间自动进行排版，图表不一定会排在当前位置，主要的目的之一是避免页面出现大段留白。

一般出现这样问题的原因是针对每个图表的文字解释和说明太少，可以增加每个图表的解释说明文字解决该问题。

(2) 我就想将图表放在当前位置，可以吗?

一般采用 LaTeX 默认的排版算法即可，在文字中 `ref` 引用图表号即可。

如确需强行将图表放在当前位置，可以采用 `[H]` 选项实现。

7.13 其他事项

以下原作者是广告时间, 插播一段广告:

- 插图的制作, 建议 `pgf`. `pgf` 的长处是源文件直接植入 $\text{T}_\text{E}\text{X}$ 文档, 管理起来非常方便. 这里有我写的一个关于初次使用 `pgf` 的帖子:

<http://bbs.ctex.org/forum.php?mod=viewthread&tid=30480>.

- 生成参考文献, 建议使用 `BibTeX`. 这里有我写的一个文档:

<http://bbs.ctex.org/forum.php?mod=viewthread&tid=26056>.

使用 `BibTeX` 做参考文献时, 借助 `EndNote` 或者 `NoteExpress`, 可以非常漂亮简单地解决 `bib` 文件的录入问题. `NoteExpress` 在校图书馆网站有正版软件提供下载. 当

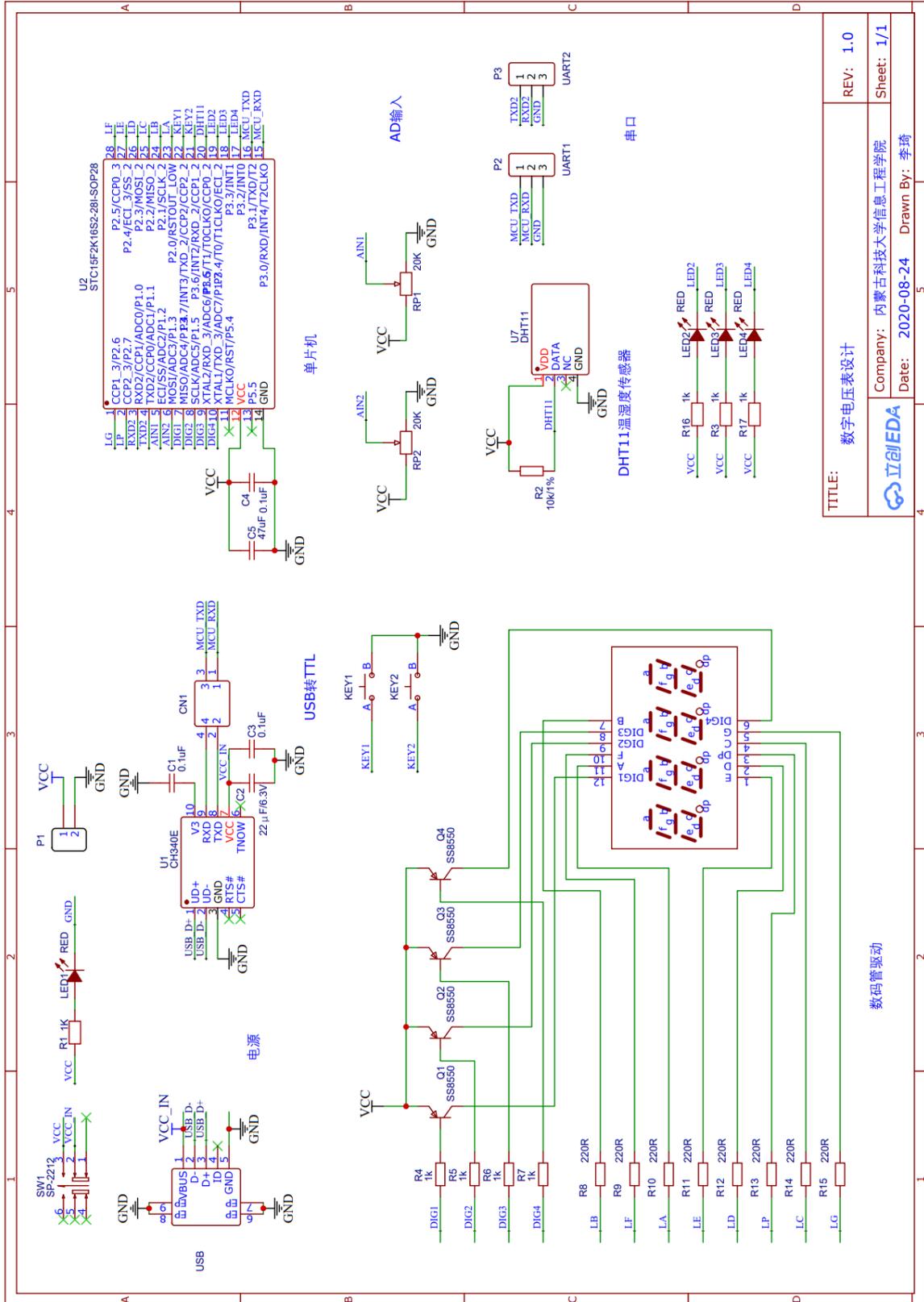
然 EndNote 本身就是 Thomson Corporation 推出的 (和 SCI 搜索引擎是同一家公司), 和多个重要文献搜索引擎有良好的功能配合. Google 学术搜索也提供了文献的 bib 格式. 录入参考文献时, 偶尔用一用 Google 学术搜索, 还可以核查或减少录入的错误, 并减少录入的工作量.

- 幻灯片的制作, 建议使用 Beamer. 这里有我写的一个模板, 谨供参考:
<http://bbs.ctex.org/forum.php?mod=viewthread&tid=27695>.

参考文献

- [1] 高阳, 陈世福, 陆鑫. 强化学习研究综述[J]. 自动化学报, 2004, 030(001): 86-100.
- [2] 高建贞, 任明武, 唐振民, 等. 路面裂缝的自动检测与识别[J]. 计算机工程, 2003(2): 154-155.
- [3] Xing E P, Ng A Y, Jordan M I, et al. Distance metric learning, with application to clustering with side-information[C]//International Conference on Neural Information Processing Systems. Massachusetts: MIT Press, 2002: 521-528.
- [4] 冯慈璋, 马西奎. 工程电磁场导论[M]. 陕西: 高等教育出版社, 2000.
- [5] 万庆祝, 陆志刚, 王科, 等. 精密谐波齿轮减速器传动误差分析[J]. 仪表技术与传感器, 2013(5): 51-54.
- [6] Silver D, Huang A, Maddison C J, et al. Mastering the game of Go with deep neural networks and tree search[J]. Nature, 2016, 529(7587): 484-489.
- [7] 成永红, 谢小军, 陈玉, 等. 气体绝缘系统中典型缺陷的超宽频带放电信号的分形分析[J]. 中国电机工程学报, 2004, 24(8): 99-102.
- [8] 吴锴, 陈曦, 王霞, 等. 纳米粒子改性聚乙烯直流电缆绝缘材料研究 (II)[J]. 高电压技术, 2013, 39(1): 8-16.
- [9] DXie123. Thesis-Template-for-XJTU[EB/OL]. 2018. <https://github.com/DXie123/Thesis-Template-for-XJTU>.
- [10] 朱少杰. 基于深度学习的文本情感分类研究[D]. 哈尔滨工业大学, 2014.
- [11] 李武军, 王崇骏, 张炜, 等. 人脸识别研究综述[J]. 模式识别与人工智能, 2006, 19(1): 58-66.
- [12] Liu Y, Yang Z, Yu Z, et al. Generative artificial intelligence and its applications in materials science: Current situation and future perspectives[J/OL]. Journal of Materiomics, 2023, 9(4): 798-816. <https://www.sciencedirect.com/science/article/pii/S2352847823000771>.
- [13] Wang X, Wang Y, Yu L, et al. Automated pavement crack detection based on multiscale fully convolutional network[J/OL]. Journal of Engineering, 2023, 2023(10). <http://dx.doi.org/10.1049/tje2.12317>.

附录 A 电路原理图



TITLE: 数字电压表设计	REV: 1.0
Company: 内蒙古科技大学信息工程学院	Sheet: 1/1
Date: 2020-08-24	Drawn By: 李琦

图 A.1 单片机驱动四位数码管显示数字电压表电路原理图

附录 B 源代码

```
1 #include <reg51.h>
2 #include <intrins.h>
3 #include "common.h"
4 #define FOSC 11059200ul
5 #define T0_H (65536-(2*FOSC)/(1000*12))/256
6 #define T0_L (65536-(2*FOSC)/(1000*12))%256
7
8 sbit COM0 = P1^0;
9 sbit COM1 = P1^1;
10 sbit COM2 = P1^2;
11 sbit COM3 = P1^3;
12 sbit TLC549_CS = P1^6;
13 sbit TLC549_DOUT = P1^5;
14 sbit TLC549_CLK = P1^7;
15 bit flag = 0;
16 bit flag1 = 0;
17 uint8_t ADcounts = 0;
18 uint8_t dispBuf[] = { 2, 0, 1, 8};
19 uint8_t code ledDeg[] = { 0xC0, 0xF9, 0xA4, 0xB0, 0x99, 0x92, 0x82, 0
    xF8, 0x80, 0x90 };
20 uint8_t bdata AdValue;
21 sbit AIN = AdValue^0;
22
23 void initSys();
24
25 uint8_t getAdValue(){
26     uint8_t i;
27     TLC549_CS = 0;
28     TLC549_DOUT = 1;
29     TLC549_CLK = 0;
30     for(i=0;i<8;i++){
31         AdValue = AdValue<<1;
32         TLC549_CLK = 1;
33         _nop_();
34         _nop_();
```

```
35     AIN = TLC549_DOUT;
36     TLC549_CLK = 0;
37     _nop_();
38     _nop_();
39 }
40 TLC549_CS = 1;
41 return AdValue;
42 }
43
44 void main(){
45     uint8_t i = 0;
46     uint8_t j = 0;
47     uint8_t ucAdValue = 0;
48     uint16_t uiAdValue = 0;
49     float fAdValue = 0.0;
50     initSys();
51     while(1){
52         if(flag1){
53             flag1 = 0;
54             ucAdValue = getAdValue();
55             //标度变换
56             fAdValue = (float)(ucAdValue*(5.0/255.0));
57             uiAdValue= fAdValue * 1000;
58             for(j = 0; j < 4; j++){
59                 dispBuf[3-j] = uiAdValue%10;
60                 uiAdValue /= 10;
61             }
62         }
63         if(flag){
64             flag = 0;
65             switch(i){
66                 case 0:
67                     COM0 = 0; COM1 = 1; COM2 = 1; COM3 = 1;
68                     break;
69                 case 1:
70                     COM0 = 1; COM1 = 0; COM2 = 1; COM3 = 1;
71                     break;
```

```
72         case 2:
73             COM0 = 1; COM1 = 1; COM2 = 0; COM3 = 1;
74             break;
75         case 3:
76             COM0 = 1; COM1 = 1; COM2 = 1; COM3 = 0;
77             break;
78     }
79     P3 = ledDeg[dispBuf[i]];
80     if(i==0)
81         P3 &= 0x7f;
82     if(++i>=4)
83         i = 0;
84     }
85 }
86 }
87
88 void initSys(){
89     TMOD &= 0xF0;
90     TMOD |= 0x01;
91     EA = 1;
92     ET0 =1;
93     TH0 = T0_H;
94     TL0 = T0_L;
95     TR0 = 1;
96 }
97
98 void timer0ISR() interrupt 1{
99     TH0 = T0_H;
100    TL0 = T0_L;
101    flag = 1;
102    if(ADcounts++>=40){
103        ADcounts = 0;
104        flag1 = 1;
105    }
106 }
```

致 谢

感谢你,感谢他和她,感谢大家.