

基于Arduino单片机的家庭绿植自动浇水系统

彭思垚, 杨艳梅

华北理工大学理学院, 河北 唐山

收稿日期: 2024年11月22日; 录用日期: 2025年2月18日; 发布日期: 2025年3月4日

摘要

利用Arduino单片机设计了一种绿植自动浇水系统, 主要包括主控模块、LCD显示模块、按键调控模块、湿度检测模块、水泵模块及WI-FI模块, 使整个系统加入物联网实现全自动化。该系统可以通过对土壤温湿度的检测来判断绿植的生长环境的现状, 即缺水还是除湿。且用户可通过手动调节土壤湿度的阈值范围, LCD显示屏显示当前土壤温湿度等信息。当土壤湿度低于设定的阈值时, 启动水泵进行抽水浇水, 当浇水后的阈值在设定的湿度范围内时, 停止浇水。WI-FI模块的加入使整个系统与物联网相连接, 方便用户在手机端监控和操作。

关键词

Arduino单片机, 绿植自动浇水系统, WI-FI模块, 物联网, 自动化

Automatic Watering System for Home Green Plant Based on Arduino Microcontroller

Siyao Peng, Yanmei Yang

College of Science, North China University of Science and Technology, Tangshan Hebei

Received: Nov. 22nd, 2024; accepted: Feb. 18th, 2025; published: Mar. 4th, 2025

Abstract

An automatic watering system for green plants is designed by using Arduino single chip microcomputer. It mainly includes main control module, LCD display module, key control module, humidity detection module, water pump module and WI-FI module, which makes the whole system join the Internet of Things to realize full automation. The system can judge the status of the growing environment of green plants through the detection of soil temperature and humidity, that is, water shortage or dehumidification. And the user can manually adjust the threshold range of soil moisture, LCD display screen displays the current soil temperature and humidity information. When the soil

moisture is lower than the set threshold, the pump is started to pump water, and when the threshold after watering is within the set humidity range, the watering is stopped. The addition of the WI-FI module tries to connect the whole system with the Internet of Things, which is convenient for users to monitor and operate on the mobile phone.

Keywords

Arduino Microcontroller, Plant Automatic Watering System, WI-FI Module, Internet of Things, Automation

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着生活水平的提高,许多居民为了美化环境、净化空气,喜欢在家中阳台上种植一些花草。但现代人们的生活节奏愈加愈快,工作越来越繁忙,经常会忘记按时给家中的绿植浇水,但植物是离不开水的。当人们出差或有事不在家时,植物的供水就会中断,植物将会面临干枯。经调查发现花草生长的问题 80%以上由绿植浇灌问题引起,不容易养的花卉会因为浇水问题而长势不好或更严重的出现枯萎甚至凋亡。

虽然目前国内外市场上存在一些自动浇水系统,但大部分自动供水灌溉系统采用虹吸方式,即这种模式只能保证不会出现干旱现象,但无法根据绿植的实际需要进行适时适量地供水。同时市面上的其他自动浇水系统需要有水龙头在外面提供的前提下才可以使用,但大部分家庭中的花草绿植一般都种植在阳台上,而在无人情况下在阳台使用水龙头十分不安全也并不方便。除此之外其他电子平台在售的智能花盆虽然种类繁多,但其中大部分都是通过花盆套叠方式实现“自动吸水”式浇水,这种方式不能根据绿植实际需求进行适时适量地浇水,实际上也仅止步于能够在规定的时间内向绿植进行补水,这两种方式基本一致,但都不能按照绿植的实际需求来进行补水灌溉。

针对上述问题,该论文研究设计了一款基于 Arduino 单片机[1]的自动浇水系统[2],该系统利用传感器检测相关数据并传达到单片机,由单片机判断是否需要绿植进行浇水或除湿,其中土壤温湿度等阈值范围可以通过手动进行调节,且可通过 LCD 显示屏显示当前土壤温湿度信息。当土壤湿度低于设定的阈值时,系统自动启动水泵进行抽水浇水,达到阈值时停止浇水。当湿度过高时启动除湿,直到湿度在设定的湿度范围内停止除湿,最终实现适时适量地自动浇水。

2. 系统设计

该系统分为:主控模块、电源模块、LCD 显示模块、按键调控模块、湿度检测模块、水泵模块。主控模块顾名思义就是对整个系统进行控制,其负责接收和发送整个系统的命令。太阳能电源模块主要是为整个系统提供电能,并且为水泵模块提供电能来让电机实现抽水功能。LCD 显示模块用来显示当前土壤湿度以及在进行阈值设定时更直观的看到设定的阈值数。湿度检测模块就是检测当前土壤中的水分含量并以电信号的形式展现出来。总体来说,该系统以单片机为控制中心,利用土壤湿度传感器对绿植土壤湿度进行检测,依靠湿度传感器检测的数据与系统预设的湿度阈值[3]进行对比。编写 Arduino 代码,实现传感器读取和水泵控制逻辑,根据湿度检测器的读数决定水泵是否需要开启浇灌,并使用可远程监

3. 硬件选择

3.1. 单片机选择

本系统中运用了经济实惠、适应性极强、兼容广泛的 Arduino UNO 控制器[4]。此类单片机[5], 配合外围器件(传感器, 控制开关, LED 屏等), 即使是只具有基础电子知识的使用者也能迅速搭建起满足个性化需求的电子项目, 不仅具备庞大的数据库支持, 而且其基础运作理念简朴易懂, 软件模式更是明晰且完备。它集成了通信系统与数字 I/O, 模拟 I/O, 更配置了 SPI 服务、通信系统, IIC 服务, 以及 UART 服务, 功能的丰富。其编程代码简单明了, 不需要复杂的逻辑链, 用户进行简单操作后即可达到预期中的效果, 也可以根据自我需要修改和开发 Arduino。单片机通过检测连接的各种传感器, 能够准确地感知到周围的环境, 并利用灯光、马达等设备, 有效地传达对环境的影响。此外, 该控制器还提供了一个易于使用的集成开发环境——IDE, 使用户拥有自主操控权和迅捷的处理速度。在整个系统中, 作为控制芯片的角色, 它有着不可替代的重要作用, 其电路设计图见图 3。

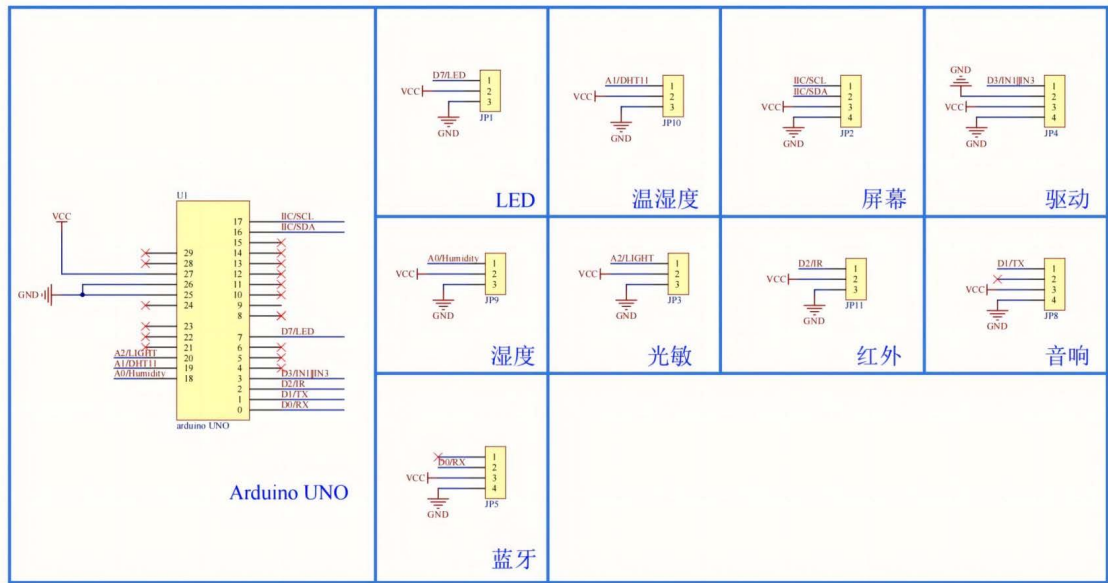


Figure 3. Circuit design diagram
图 3. 电路设计图

3.2. 湿度传感器模块

湿度传感器[6]是一种价格低廉、灵敏度高的电子设备, 它可以用来测量空气中的湿度。该系统选用了 AM2302 型湿敏电容型数码温湿控制模组[7], 该传感器采用了一种基于 8 位微控制器的电容感湿元件和温度测量单元, 通过更准确的湿度数据让系统实时调控当前水泵状态并快速传达数据, 通过数字式输出。连接湿度传感器到系统中, 实时监测花盆内土壤湿度, 并通过电信号展示测量数据。

3.3. 显示模块

本系统配备 LCD1602 液晶显示模块, 它是利用液晶的物理特性, 通过电压对其显示区域进行控制, 从而显示字符和简单图形, 允许用户实时查看时间、植物名称、用户预先设定好的参数、湿度数据以及产品的运行状态, 具备有灌溉或待机模式。用户可以通过可视化的测试数据随时了解花盆的状态, 同时也可以及时调整阈值数据。

3.4. 水泵控制模块

该系统使用继电器作为水泵系统的开关, 通过电流控制衔铁和触电簧片, 实现电路的开合, 通过对电机的控制实现向花盆供水。可以将此系统视为使用低电压来控制高压的“断路器”。水泵系统中有电磁继电器, 它主要由线圈、衔铁、铁芯和触电簧片组成, 当电流流过线圈时, 由于电压产生电磁效应并产生磁力, 衔铁由于磁力接触电路, 形成闭路, 电机 **M** 启动, 水泵工作开始抽水, 否则衔铁将始终与电路断开, 无法形成闭路, 没有电流, 不会产生磁力, 水泵就不会运转。

3.5. 音响模块

系统中的报警系统主要是通过音响完成的。当湿度传感器检测到湿度小于用户所设阈值, 水泵开始抽水, 水泵结束抽水时均有报警。另外, 用户修改参数, 设置相关功能时也会通过音响模块播报。音响系统可根据用户个人喜好通过移动端设置是否播报, 播报音量以及具体播报内容。

3.6. WIFI 模块

系统所用的 ESP 系列 WIFI 模块可以实现将该系统接入互联网, 同时也可精准上传花盆内各种数据。用户可通过连接 APP 来对植物生长环境及生长状况进行远程监控, 同时用户还可以从手机上远程控制浇水、照明等参数。当出现异常情况时系统会自动传达指令为用户在软件上提醒, 便于用户更及时接受和检测一切信息。

3.7. 按键调控模块

按键调控模块是一种利用按键的操控模块[8], 通过按键简化人机互动的复杂操作, 按键可准确传达信号, 以便系统做出指令, 并且可以实现人机交互, 即当接入按键到系统中, 该按键可以调节数据阈值来达到监测作用, 通过调节数据阈值纠正错误, 实时调节纠正。

3.8. 电源模块

电源模块主要选择太阳能电源模块给系统供电, 利用太阳能模块转化太阳能为电能给各个元件供电, 以及给水泵供电来实现各个模块的正常工作。同时配备有一块电池白天太阳能模板供电的同时给电池充电, 夜晚没有太阳能由电池供电, 来保证系统不会断电。

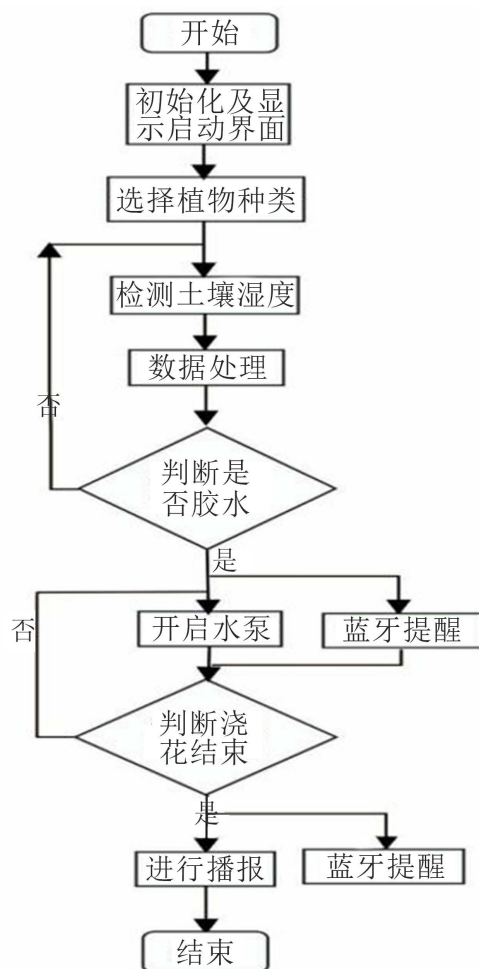
4. 软件设计

安装 Arduino IDE 开发环境, 并下载所需的库, 如 WIFI 库、TFT 库。连接 Arduino IDE 控制板, 并编程烧如代码, 完成硬件与软件的集成。借助 Arduino 嵌入式平台的 WIFI 模块, 该系统采集的温湿度等各类数据可实时同步到用户的移动端 App, 方便用户对植物进行实时监控。同时当设备上线后, APP 页面会及时更新采集的数据, 包括温湿度含氧量等状态, 用户可通过点击相应控件实现远程控制。同时, 系统加入的音响模块可对各类情况进行播报或执行用户其他指令。

系统流程图

本系统软件结构由多个程序部分组成, 不同部分拥有不同功能。关于程序流程, 首先通过初始化启动显示界面, 在用户端选择植物种类, 系统开始监测土壤湿度, 并为其在数据库里选择合适湿度数据阈值, 通过数据处理判断是否为其浇水, 如无需浇水则系统继续实时监测土壤湿度, 若需浇水则通过蓝牙音响进行提醒, 并开启水泵, 之后持续监测土壤湿度并判断是否浇花结束, 利用音响提醒并进行播报。

程序流程图如图 4 所示:



注: 源代码请见附录。

Figure 4. Program flow chart

图 4. 程序流程图

5. 项目运作方式及应用效果

湿度检测器实时检测绿植的土壤湿度, 并将湿度值传输给 Arduino 主控板。

① Arduino 主控板根据湿度传感器的读数, 判断是否需要开启水泵浇水。

② 如果土壤湿度低于设定阈值, 则 Arduino 主控板通过控制水泵开启, 将适量的水注入绿植的土壤中, 从而实现自动浇水的功能。

③ 同时, ESP 系列 WIFI 模块可以将设备连接到互联网, 用户可以通过手机等终端设备, 实现远程控制和监测设备的状态和数据。

④ 如果设备发生故障或者出现异常情况, 用户可以通过维护和优化来保持设备的正常运行状态。

开机启动, 读取土壤温湿度等传感器的数值, LCD 显示器显示对应数值, 主控模板根据设定阈值判断是否开启水泵进行浇水。接入 WiFi 后系统将各种数据传入手机端 APP, 通过本自动浇水系统, 用户可以实现对家庭绿植的智能管理, 减轻了用户的日常工作量, 同时也可以保证植物的健康生长。

6. 项目优势

1) 专一针对性:

该研究是专为家庭绿植的自动浇水而设计的, 传感器选择和功能设计都是针对绿植生长所需的水分和养分。相对于一般的自动浇水系统, 本项目的专一针对性更加突出, 可以更好地满足绿植的需求, 从而提高生长质量。

2) 实时观测数据:

该设计集成了温湿度检测器等传感器, 可以实时监测绿植的生长环境, 对土壤温湿度等参数进行实时测量并传输给主控板, 通过显示屏显示, 可以方便用户随时观测数据, 及时了解绿植的生长状态和环境变化, 从而采取措施保障绿植的健康成长。

3) 成本低:

该设计采用了一些经济实惠的传感器和元器件, 如 Arduino 主控板和水泵, 以及便宜但性能很好的 ESP 系列 WiFi 模块, 成本相对较低, 对于一般家庭用户而言, 可以更好地满足他们的需求。

4) 智能化:

该设计加入了物联网使其实现对家庭绿植的全自动化控制, 大大减轻了用户的负担, 同时用户还可以通过远程控制来监测设备的状态和数据, 实现智能化的管理和维护, 提高了生活的品质。同时, 智能化的设计可以根据不同绿植的生长环境需求进行精准养护, 不仅利于绿植的生长, 还符合节能环保的理念。

5) 发展前景:

国内外小型自动浇水系统研发较少, 该研究弥补了市场的空白, 具有巨大发展前景。

7. 研究结论

该研究基于 Arduino 单片机原理成功研发了自动浇水系统, 该系统结合移动端 APP, 实现了对温湿度、酸碱度、光照等关键参数的实时监测与智能语音控制, 展现了高度的实用性和创新性。通过湿度检测器监测土壤湿度与单片机控制系统实现对不同部位的土壤湿度监测和自动浇水, 更有利于植物的生长。且此绿植自动浇水系统制造成本低廉, 适合小型工厂及日常生活使用, 并且操作简便实用。系统所用器件兼具体积小、成本低、精度高的特点, 显著提升了用户体验并降低了成本投入。

此外互联网的发展可谓十分迅捷, 各个行业几乎都能与互联网相结合来创造新价值, 当然家庭绿植自动浇水方面也不例外。通过线上精准观测加之帮人们自动浇水、适量浇水观测生长环境、手机端远程操控等功能的扶持, 功能齐全, 为人们种养绿植提供了便利。通过互联网的宣传, 可吸引一些人去尝试并使用本系统, 从而带来潜在的消费市场。另外, 目前市场上的各种自动浇水系统也都只是单纯的浇水, 并没有如此丰富的功能, 我们的产品同时填补了市场的空白。展望未来, 我们将持续优化系统性能, 拓展更多功能, 以满足用户日益增长的需求, 推动物联网技术在智能家居领域的深入应用。

基金项目

华北理工大学大学生创新创业训练计划项目(Y2023161); 华北理工大学教育教学改革研究与实践项目(T-ZJ2218)。

参考文献

- [1] 卢筱蕾, 马薇, 孙欣然. 基于 Arduino 的智能浇水花盆[J]. 电子世界, 2020(16): 171-173.
- [2] 曾滢. 植物自动浇水系统设计[J]. 电子测试, 2018, 29(15): 31-42.
- [3] 张亚峰, 郭霞云. 基于 STM32 和 OneNET 云平台的智能花盆远程操控系统设计[J]. 工业控制计算机, 2022,

35(12): 151-152+155.

- [4] 李明, 任鹏, 孔德龙, 等. 基于 Arduino 的自动浇花机[J]. 电气时代, 2019(12): 72-74.
- [5] 刘强强, 李海潮, 王爽, 等. 基于单片机控制的园林智能浇水系统设计[J]. 模具制造, 2023, 23(10): 190-192.
- [6] 方泽鹏, 黄双萍, 陈仲涛. 基于单片机的花盆土壤湿度控制系统设计[J]. 现代农业装备, 2013(4): 41-45.
- [7] 朱嵘涛, 徐爱钧. 单总线传感器 AM2302 温湿度测控系统原理及应用[J]. 单片机与嵌入式系统应用, 2016, 16(4): 46-48.
- [8] 夏贵辉, 吴燕, 丁春发, 等. 基于单片机的智能水杯控制系统设计[J]. 电子制作, 2023, 31(23): 33-37.

附 录

(注: 由于 lcd 和 WIFI 模块程序过于庞大不便演示, 故在此只展示自动浇水主程序)

```
#include <Wire.h>
#include "LiquidCrystal_I2C.h"
#include "QhDHT11.h"
#define DHT11_PIN A1
#define MOTOR_INA1 3
#define MOTOR_INA2 5
#define LED_RED_PIN 7
#define LOW_TH 30
#define HIGH_TH (LOW_TH+5)

int pump_sta = 0; //0 代表水泵已经关闭 1 代表水泵已经开启
LiquidCrystal_I2C lcd (0x27, 16, 2);
QhDHT11 qhDHT;
int wetPin = A0;
long int wetValue = 0;

void setup (){
    Serial.begin (9600); c
    lcd.init ();                // initialize the lcd
    lcd.backlight ();
    lcd.setCursor (2,0); // go to start of 2nd line
    lcd.print ("Hello, world!");
    lcd.setCursor (4,1); // go to start of 2nd line
    lcd.print ("QHEBOT")
    pumpInit ();

    pinMode (LED_RED_PIN, OUTPUT);
    digitalWrite (LED_RED_PIN, LOW);
    Serial.println ("Hello, QHEBOT");
}

void loop (){
    getSensor ();                //得到传感器数据
    pump_judge ();              //对数据进行判别, 看是否需要打开水泵
    update_lcd1602 ();          //更新 1602 液晶数据
    delay (500);
}

void pump judge (){
```

```
    if (wetValue > HIGH_TH){
        closePump ();
    }else if (wetValue < LOW_TH){
        openPump ();
    }
}

void getSensor(){
    wetValue = analogRead (wetPin);
    wetValue *= 100;
    wetValue = wetValue/1023;

    qhDHT.getTHValue (DHT11_PIN);

    Serial.print (wetValue);
    Serial.print (",\t");
    Serial.print (qhDHT.humidity, 1);
    Serial.print (",\t");
    Serial.println (qhDHT.temperature, 1);
}

void update_lcd1602(){
    lcd.setCursor (1, 0);
    lcd.print ("TH");
    lcd.print (LOW_TH);
    lcd.print ("P");
    if (pump_sta == 1){
        lcd.print ("ON      ");
    }else{
        lcd.print ("OFF      ");
    }
    lcd.setCursor (0, 1);
    lcd.print ("T");
    if (qhDHT.temperature <= 99){
        lcd.print (qhDHT.temperature, 0);
    }else{lcd.print(99, 0);}
    lcd.print ("H");
    if (qhDHT.humidity <= 99){
        lcd.print (qhDHT.humidity, 0);
    }else{
        lcd.print (99, 0);
    }
}
```

```
    }  
    lcd.print("W");  
    lcd.print(wetValue);  
    lcd.print ("  ");  
}  
void pumpInit(){  
    pinMode (MOTOR_INA1, OUTPUT);  
    pinMode (MOTOR_INA2, OUTPUT);  
    analogWrite (MOTOR_INA1, 0);  
    analogWrite (MOTOR_INA2, 0);  
}  
void openPump (){  
    pump_sta = 1;  
  
    analogWrite (MOTOR_INA1, 255);  
    analogWrite (MOTOR_INA2, 0);  
    digitalWrite (LED_RED_PIN, HIGH);  
  
}  
  
void closePump (){  
    pump_sta = 0;  
    analogWrite (MOTOR_INA1, 0);  
    analogWrite (MOTOR_INA2 , 0);  
    digitalWrite (LED_RED_PIN, LOW);  
}
```