



دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

عنوان:

پروژه گلدان هوشمند

نگارش:

امیر محمد ابوئی، بنیامین قاسمی نیا

استاد :

دکتر اجلالی

نیم سال اول ۱۴۰۱ - ۱۴۰۰

سلام

سپاس

چکیده

کلیدواژه‌ها:

فهرست مطالب

۱۱	۱ مقدمه
۱۱	۱-۱ تیتربندی‌های کلیدی درمورد پروژه
۱۱	۲-۱ شرح مسئله
۱۲	۳-۱ راه‌کار پیشنهادی ما
۱۲	۴-۱ ساختار گزارش
۱۳	۲ نحوه پیکربندی سخت‌افزاری
۱۳	۱-۲ بلوک دیاگرام سیستم
۱۴	۲-۲ عملکرد و ارتباط کلی بخش‌های مختلف
۱۴	۱-۲-۲ سنسور رطوبت سنج
۱۴	۲-۲-۲ ماورای صوت
۱۴	۳-۲ ماژول شیر برقی
۱۵	۴-۲ ماژول رله
۱۵	۱-۴-۲ برد Arduino D1-Rev1
۱۵	۲-۴-۲ برد Arduino Uno
۱۵	۳-۴-۲ آداپتور

۱۶	۳ سخت افزار مورد نیاز
۱۶	۱-۳ آردوینو UNO
۱۶	۱-۱-۳ امکانات و ویژگی ها
۱۹	۲-۳ ماژول وای فای ESP8266EX
۱۹	۱-۲-۳ امکانات کلیدی
۲۱	۲-۲-۳ ویژگی های جزئی
۲۲	۳-۳ آردوینو D1 نسخه Rev1
۲۳	۴-۳ ماژول رطوبت سنج
۲۳	۱-۴-۳ سنسور YL-69
۲۴	۲-۴-۳ سنسور HR202
۲۴	۳-۴-۳ سنسور XH-M214
۲۴	۵-۳ شیربرقی
۲۴	۶-۳ رله
۲۵	۷-۳ آلتراسونیک
۲۵	۸-۳ آداپتور
۲۶	۴ رابط کاربری
۲۶	۱-۴ نیازمندی ها
۲۶	۲-۴ نحوه پیاده سازی
۲۸	۵ نحوه پیاده سازی فیزیکی
۲۸	۱-۵ اتصال ماژول رطوبت سنج
۲۹	۲-۵ اتصال ماژول آلتراسونیک
۳۱	۳-۵ اتصال ماژول شیربرقی

۳۲	۴-۵ اتصال رله
۳۳	۵-۵ اتصال دو بورد اصلی به یک دیگر
۳۴	۶-۵ جمع بندی
۳۵	۶ نحوه پیاده سازی نرم افزاری
۳۵	۱-۶ نحوه اتصال مشتری
۳۵	۲-۶ تنظیمات اولیه و متغیرها
۳۶	۳-۶ تعیین فاصله ی سطح آب
۳۷	۴-۶ نحوه کار سیستم در هر گردش
۳۷	۵-۶ ارتباط با مشتری
۳۷	۱-۵-۶ فعال سازی سیستم
۳۸	۲-۵-۶ دریافت حالت سیستم
۳۸	۳-۵-۶ دریافت حالت شیر
۳۹	۴-۵-۶ دریافت مقدار آستان رطوبت
۳۹	۵-۵-۶ بروزرسانی وضعیت مشتری
۴۰	۶-۶ چاپ متغیرها در ترمینال
۴۰	۷-۶ نحوه کار بورد آردینو UNO
۴۰	۱-۷-۶ ارسال کد از طرف مدار اصلی
۴۱	۲-۷-۶ نحوه کار در هر گردش
۴۲	۷ جمع بندی
۴۳	۱- مطالب تکمیلی

فهرست شکل‌ها

- ۱-۲ بلوک دیاگرام طراحی شده برای گلدان هوشمند. ۱۳
- ۱-۳ برد آردوینو UNO. ۱۶
- ۲-۳ نمای بالا از برد Arduino Uno R3. ۱۸
- ۳-۳ برد آردوینو D1 نسخه Rev1 با تراشه ESP8266. ۲۲
- ۴-۳ سنسور رطوبت سنج YL-69 با خروجی دیجیتال و آنالوگ. ۲۳
- ۵-۳ شیربرقی سلونوئید ۱۲ ولتی. ۲۴
- ۶-۳ رله مورد نیاز برای اتصال به شیر برقی. ۲۴
- ۷-۳ ماژول آلتراسونیک HY-SRF05. ۲۵
- ۱-۴ رابط کاربری برنامه گلدان هوشمند در نرم‌افزار Blynk. ۲۷
- ۱-۵ تصویر ماژول رطوبت‌سنج در خاک. ۲۸
- ۲-۵ جاسازی ماژول آلتراسونیک در مخزن. ۲۹
- ۳-۵ نمای بالای ماژول آلتراسونیک به همراه قیف ورودی آب. ۳۰
- ۴-۵ اتصال ماژول شیربرقی به مخزن آب. ۳۱
- ۵-۵ رله تک کانال بکاربرده شده در مدار. ۳۲
- ۶-۵ آداپتور ۱۲ ولتی مورد نیاز برای رله. ۳۲

- ۷-۵ اتصال دو برد برای برقراری ارتباط سریال. ۳۳
- ۸-۵ شمای کلی طراحی و مدار پیاده‌سازی شده. ۳۴

فهرست جدول‌ها

فصل ۱

مقدمه

۱-۱ تیتربندی‌های کلیدی درمورد پروژه

حوزه کاربرد پروژه: خانه هوشمند^۱ و به‌طور خاص اتوماتیک‌سازی خانه^۲

هدف پروژه: آبیاری خودکار گیاهان خانگی

۲-۱ شرح مسئله

این نکته شناخته شده است که نگهداری از گیاه خانگی می‌تواند علاوه بر سلامت فیزیکی به دلیل بهبود کیفیت هوا، اثرات مثبت ذهنی و روانی نیز بر جو خانه بگذارد. بدین جهت انسان از مدت‌ها قبل اقدام به پرورش و نگهداری گیاه‌های تزئینی به‌طور خانگی گرفته است؛ در دنیای امروزی ممکن است به دلیل مشغله‌های ذهنی زیاد و درگیری‌های کاری، رسیدگی منظم به گیاه‌ها را برای صاحبش دشوار سازد.

بنابراین هر روشی که بتواند برای نگهداری گیاهان خانگی به صاحب آن کمک بکند، از اهمیت بالایی برخوردار است. به‌طور خاص توجه این پروژه بر جانب آبیاری منظم گیاه‌های خانگی است. آبیاری فراتر از حد درست می‌تواند موجب پوسیدگی ریشه گیاهان شود و آبیاری دیر به دیر آن‌ها موجب خشک شدن گیاه.

¹Smart Home

²Home Automation

۱-۳ راهکار پیشنهادی ما

در این پروژه هدف داریم راهکاری برای خودکار سازی و هوشمندسازی فرآیند آبیاری گیاه بپردازیم، همچنین این امکان را برای پرورش‌دهنده گیاه فراهم سازیم تا با راحتی بیشتری از وضعیت گیاه(های) خود مطلع شود. در ابتدا راهحل پیشنهادی را به طور کلی مطرح می‌کنیم سپس به‌طور دقیق قطعاتی که مورد استفاده خواهند بود را ذکر می‌کنیم.

راهحل پیشنهادی ما بدین‌صورت است که با قراردادن یک سنسور در گلدانی که گیاه در آن نگهداری می‌شود، می‌توان میزان رطوبت خاک آن را در نظر گرفت. در ادامه بُرد آردوینو^۳ که به این سنسور متصل است، از طریق اینترنت به یک اپ برای گوشی‌های اندروید متصل می‌شود و اطلاعات مربوط به رطوبت موجود در خاک گیاه را برای کاربر به نمایش می‌گذارد. تا همینجا این کار می‌تواند به پرورش‌دهنده گیاه برای رصد وضعیت گیاهش کمک کند. قابلیت دیگری که این سیستم دارد آبیاری خودکار است. برای این کار یک موتور محرک برای شیر هوشمند در معرض گلدان قرار داده‌ایم که به برد متصل است. سیستم با توجه به میزان رطوبت خاک، اقدام به اضافه کردن آب به آن می‌کند. کاربر نیز خود امکان آبیاری در زمان دل‌خواه را با استفاده از اپ دارد. آب مورد استفاده برای آبیاری گیاه از یک مخزن ثابت تهیه می‌شود؛ در آن مقصد یک سنسور ماورای صوت وجود دارد که سطح آب مخزن را می‌سنجد تا در صورت اتمام، کاربر مطلع شود.

۱-۴ ساختار گزارش

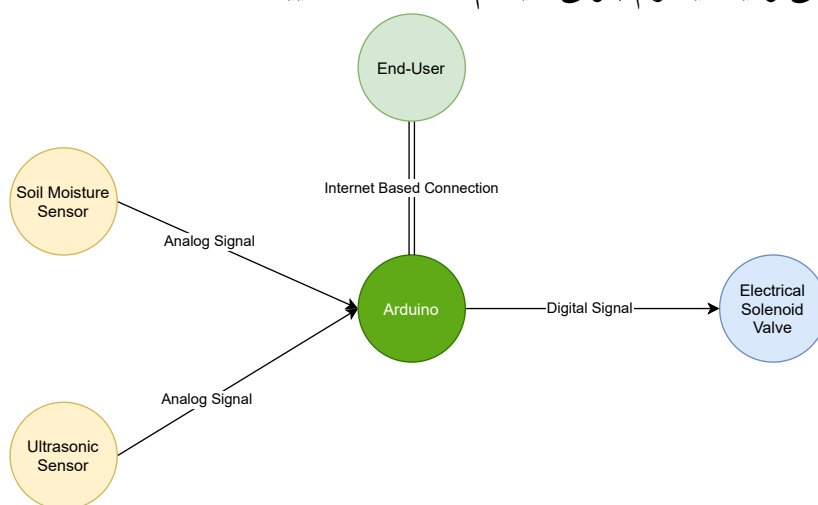
³Arduino

فصل ۲

نحوه پیکربندی سخت‌افزاری

۱-۲ بلوک دیاگرام سیستم

در شکل ۱-۲ می‌توانید دیاگرام بلوکی سیستم را مشاهده نمایید



شکل ۱-۲: بلوک دیاگرام طراحی شده برای گلدان هوشمند.

۲-۲ عملکرد و ارتباط کلی بخش‌های مختلف

۱-۲-۲ سنسور رطوبت سنج

همان‌طور که از نامش پیداست، این سنسور در خاکی که گیاه در آن قرار گرفته، کار گذاشته می‌شود. در بخش ۳-۴ مفصل‌تر در مورد سخت‌افزار سنسور توضیح داده‌ایم. کاری که این سنسور انجام می‌دهد این است که مقدار خشک‌بودن خاک را توسط دو شاخه‌اش اندازه گرفته و آن را به شکل سیگنال آنالوگ، برای برد ارسال می‌کند. این مقادیر دریافتی، با محدوده‌ای که برای خشکی تعیین می‌کنیم سنجیده می‌شود و در صورتی که خشکی از سطحی که انتظار داشتیم بیشتر بود و سیستم در حالت اتوماتیک بود، آبیاری گیاه صورت می‌گیرد.

۲-۲-۲ ماورای صوت

این ماژول، با قرارگرفتن در بالای مخزن آب، سیگنال‌های ماورای صوت ارسال کرده و بازتاب آن‌ها را دریافت می‌کند. با داشتن اختلاف زمانی بین سیگنال ارسالی و دریافتی می‌توان عمق آب را تشخیص داد. در عمل، سیگنال توسط کد برنامه فرستاده می‌شود و سیگنال بازگشتی به شکل آنالوگ به برد داده می‌شود؛ سپس اختلاف زمانی این دو سنجیده می‌شود تا از روی آن عمق آب محاسبه گردد.

۳-۲ ماژول شیر برقی

این ماژول کنترل شلنگ خروجی از مخزن آب را برعهده دارد. عملکرد این ماژول با ولتاژ ۱۲ ولت است و در دو حالت باز یا بسته می‌تواند قرار بگیرد. با توجه به میزان رطوبت یا دستور ارسالی توسط کاربر، با کمک رله‌ای که میان برد و این ماژول قرار دارد، باز و بسته‌شدن آن کنترل می‌شود و آبیاری صورت می‌گیرد.

۴-۲ ماژول رله

عملکرد این ماژول این است که سیگنال‌های ۰ یا ۵ ولتی که از برد آردوینو ارسال شده‌اند را به منظور قطع یا وصل کردن ماژول شیر برقی به‌کار بگیرد. در صورتی که سیگنال ارسالی ۵ ولت باشد، رله باعث می‌شود شیر برقی شروع به کار کند و وقتی سیگنال ارسالی ۰ ولت باشد، آن را از اتصال خارج می‌کند.

۱-۴-۲ برد Arduino D1-Rev1

این برد اصلی سیستم است و همگی سیگنال‌های دریافتی و ارسالی سیستم در این برد کنترل می‌شوند، به استثنای سیگنال‌های ارسالی به رله مربوط به شیر برقی. در عوض سیگنال‌های ارسالی به رله، توسط یک برد دوم صورت می‌گیرد. خود آن برد، تنها یک سیگنال ۰ و ۱ توسط این برد دریافت می‌کند که مشخص می‌کند سیگنال ارسالی به رله چه باشد. به جز آن، همگی سیگنال‌های ورودی توسط ماژول رطوبت‌سنج، ماژول ماورای صوت، و سیگنال‌های ارسالی توسط اپ‌کاربر در همین برد پردازش شده و سیگنال‌های خروجی مربوطه به آن‌ها ارسال می‌شود.

۲-۴-۲ برد Arduino Uno

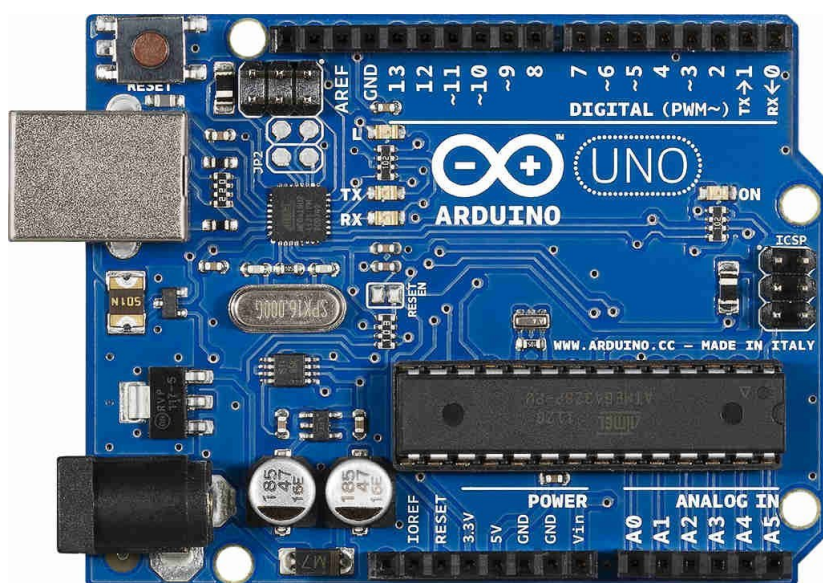
این برد ثانویه، صرفاً برای ارسال سیگنال‌های باز یا بسته‌شدن برای رله متصل به شیر برقی است. خود این برد از برد اصلی یک سیگنال ۰ یا ۱ دریافت می‌کند که اگر ۰ بود مشخص می‌کند که سیگنال ارسالی به رله، ولتاژ ۰ باشد و اگر ۱ بود، سیگنال ارسالی ولتاژ ۵ باشد.

۳-۴-۲ آداپتور

فصل ۳

سخت افزار مورد نیاز

۱-۳ آردوینو UNO



شکل ۱-۳: برد آردوینو UNO.

۱-۱-۳ امکانات و ویژگی ها

• پردازنده: ATmega328P

– حافظه

* CPU AVR تا ۱۶ مگاهرتز

* ۳۲ کیلوبایت کش

* ۲ کیلوبایت SRAM

* ۱ کیلوبایت EEPROM

— امنیت

* Power On Reset (POR)

* Brown Out Detection (BOD)

— امکانات جانبی

* $2 \times 8\text{-bit}$ تایمر/شمارنده با ثبات تناوب^۱ اختصاصی و کانال های مقایسه ای^۲

* $1 \times 16\text{-bit}$ تایمر/شمارنده با ثبات تناوب اختصاصی، گرفتن ورودی و کانال های

مقایسه ای

* $1 \times \text{USART}$ با تولیدکننده تناوبی نرخ-باد^۳ و تشخیص دهنده ابتدای قاب^۴

* $1 \times \text{controller/peripheral}$ رابط موازی کاربری پرفرمانس^۵

* $1 \times \text{Dual mode controller/peripheral I2C}$

* $1 \times \text{Analog Comparator (AC)}$ با ورودی معیار مقیاس پذیر^۶

* Watchdog Timer با اسیلاتور مجزای بر-تراشه

* شش کانال PWM^۷

* وقفه^۸ و wake-up هنگام تغییر پین

— پردازنده ATmega16U2

* پردازنده ۸ بیتی AVR تحت RISC

* حافظه

¹Period Register

²Compare Channel

³Baud-Rate

⁴start-of-frame

⁵Serial Peripheral Interface (SPI)

⁶Scalable reference input

⁷Pulse-Width Modulation

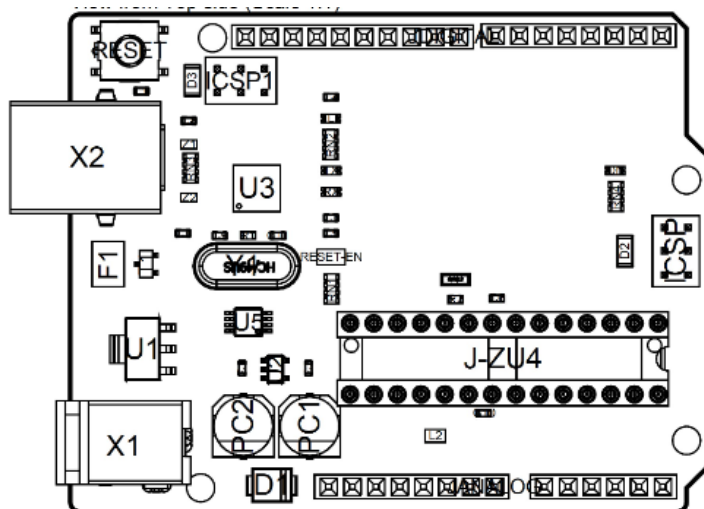
⁸Interrupt

- ۱۶ کیلوبایت ISP Flash
- ۵۱۲ بایت EEPROM
- ۵۱۲ بایت SRAM
- رابط کاربری debugWIRE برای دیباگ کردن و برنامه ریزی بر-تراشه

– انرژی

* ۵/۵-۲/۷ ولت

در شکل ۲-۳ می توانید نمای بالایی از برد و اجزای آن را ببینید.



Board topology

Ref.	Description	Ref.	Description
X1	Power jack 2.1x5.5mm	U1	SPX1117M3-L-5 Regulator
X2	USB B Connector	U3	ATMEGA16U2 Module
PC1	EEE-1EA470WP 25V SMD Capacitor	U5	LMV358LIST-A.9 IC
PC2	EEE-1EA470WP 25V SMD Capacitor	F1	Chip Capacitor, High Density
D1	CGRA4007-G Rectifier	ICSP	Pin header connector (through hole 6)
J-ZU4	ATMEGA328P Module	ICSP1	Pin header connector (through hole 6)
Y1	ECS-160-20-4X-DU Oscillator		

شکل ۲-۳: نمای بالا از برد Arduino Uno R3

۲-۳ مازول وای فای ESP8266EX

۱-۲-۳ امکانات کلیدی

- پشتیبانی از 802.11 b/g/n
- پشتیبانی از 802.11 n تا سقف ۷۲/۲Mbps
- یکپارچه سازی^۹
- ۲ رابط کاربری مجازی وای فای
- مونیتور خودکار beacon (TSF)^{۱۰}
- پشتیبانی از زیرساختار BSS Station mode، SoftAP mode و Promiscuous mode

^۹Defragmentation

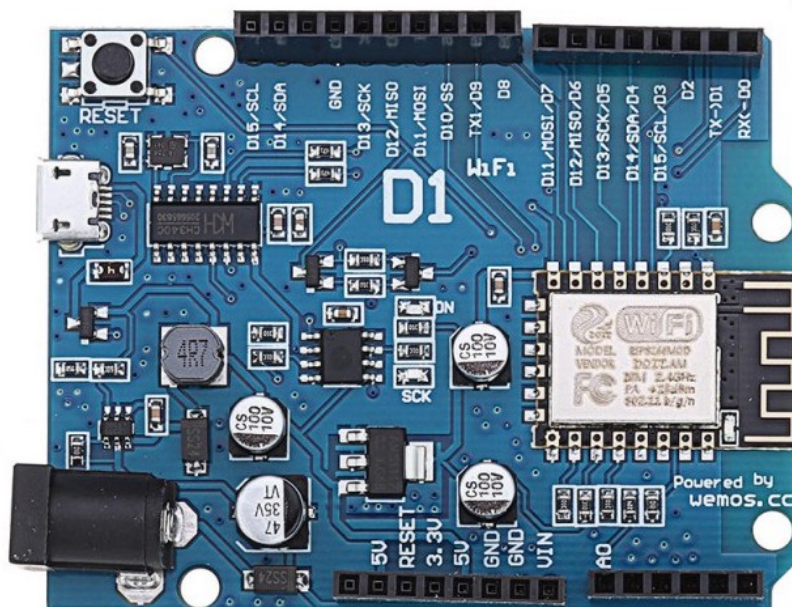
^{۱۰}Timing synchronization function

۲-۲-۳ ویژگی های جزئی

دسته بندی	آیتم	پارامترها
Wi-Fi	Certification	Wi-Fi Alliance
	Protocols	802.11 b/g/n (HT20)
	Frequency Range	2.4 GHz ~ 2.5 GHz (2400 MHz ~ 2483.5 MHz)
	TX Power	802.11 b: +20 dBm
		802.11 g: +17 dBm
		802.11 n: +14 dBm
	Rx Sensitivity	802.11 b: ~ 91dbm (11 Mbps)
		802.11 g: ~ 75dbm (54 Mbps)
		802.11 n: ~ 72 dbm (MCS7)
	Antenna	PCB Trace, External, IPEX Connector, Ceramic Chip
Hardware	CPU	Tensilica L106 32-bit processor
	Peripheral Interface	UART/SDIO/SPI/I2C/I2S/IR Remote Control
		GPIO/ADC/PWM/LED Light & Button
	Operating Voltage	2.5V ~ 3.6V
	Operating Current	Average value: 80 mA
	Operating Temperature	Range ~ 40°C ~ 125°C
	Package Size	QFN32-pin (5mm × 5mm)
	External Interface	-
Software	Wi-Fi Mode	Station/SoftAP/SoftAP+Station
	Security	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download/OTA (via network)
	Software Development	Supports Cloud Server Development/Firmware and SDK for fast on-chip programming
	Network Protocols	IPv4, TCP/UDP/HTTP
	User Configuration	AT Instruction Set, Cloud Server, Android/iOS App

۳-۳ آردوینو D1 نسخه Rev1

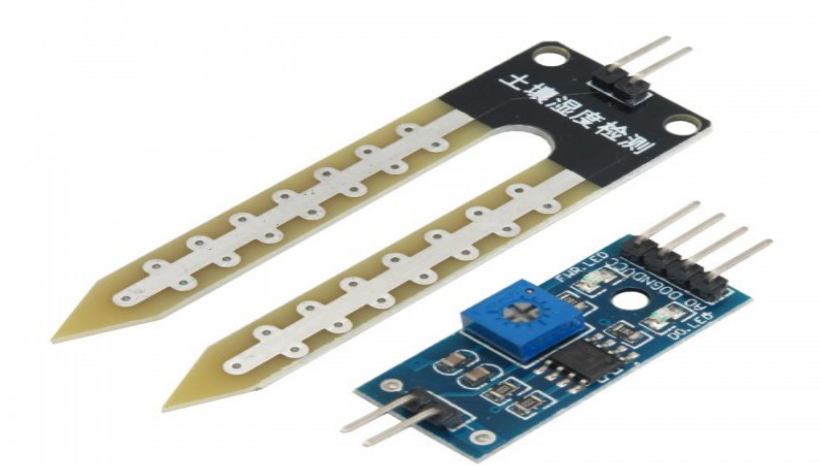
این برد، همان برد Arduino Uno است که ماژول وای فای به آن متصل است.



شکل ۳-۳: برد آردوینو D1 نسخه Rev1 با تراشه ESP8266.

۳-۴ ماژول رطوبت سنج

۳-۴-۱ سنسور YL-69



شکل ۳-۴: سنسور رطوبت سنج YL-69 با خروجی دیجیتال و آنالوگ.

۲-۴-۳ سنسور HR202

۳-۴-۳ سنسور XH-M214

۵-۳ شیر برقی



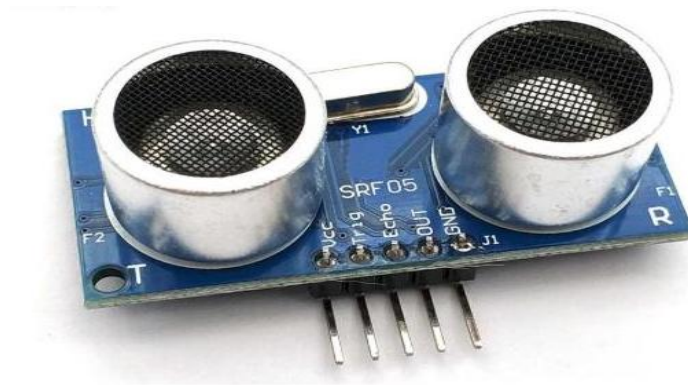
شکل ۵-۳: شیر برقی سلونوئید ۱۲ ولتی.

۶-۳ رله



شکل ۶-۳: رله مورد نیاز برای اتصال به شیر برقی.

۷-۳ آلتراسونیک



شکل ۷-۳: ماژول آلتراسونیک HY-SRF05.

۸-۳ آداپتور

فصل ۴

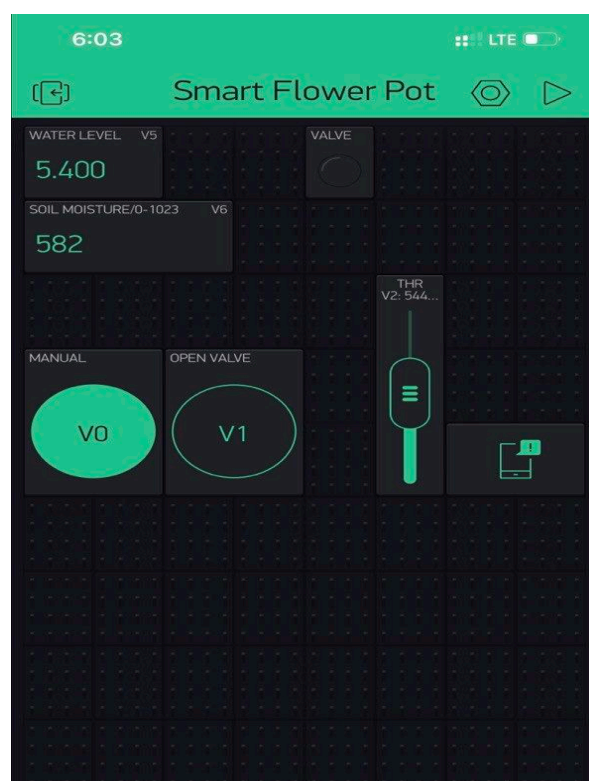
رابط کاربری

۴-۱ نیازمندی‌ها

•

۴-۲ نحوه پیاده‌سازی

استفاده از پلتفرم Blynk.

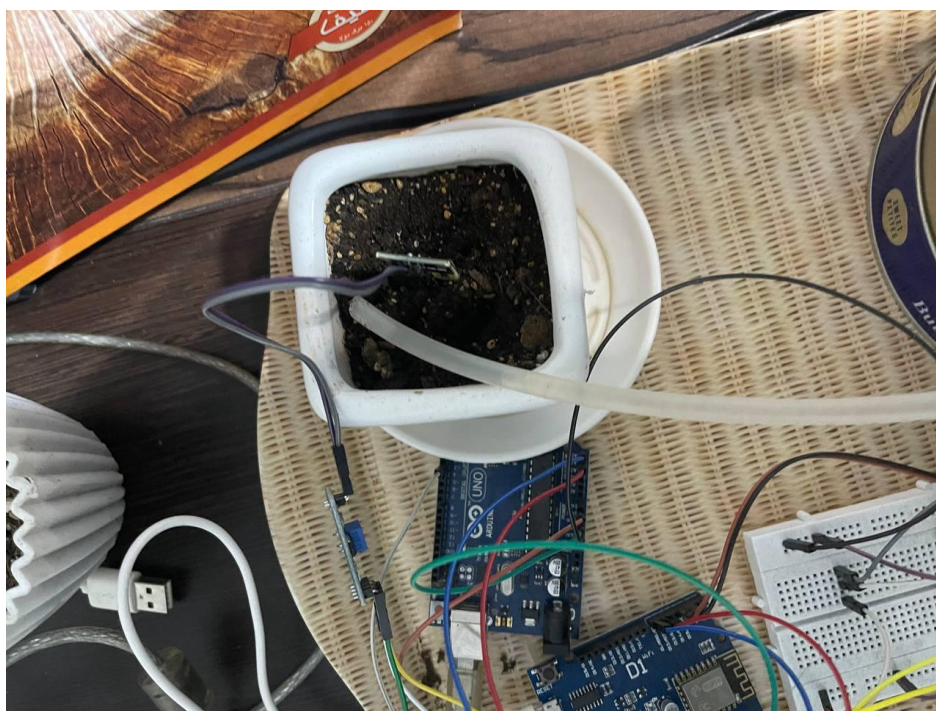


شکل ۴-۱: رابط کاربری برنامه گلدان هوشمند در نرم افزار Blynk.

فصل ۵

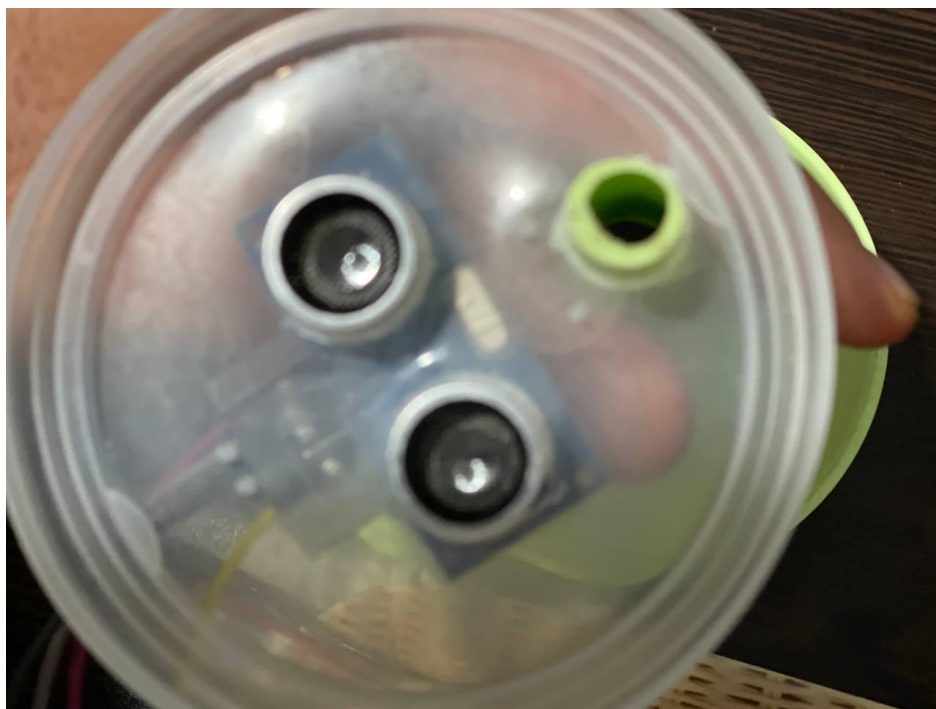
نحوه پیاده‌سازی فیزیکی

۵-۱ اتصال ماژول رطوبت سنچ



شکل ۵-۱: تصویر ماژول رطوبت سنچ در خاک.

۵-۲ اتصال ماژول آلتراسونیک



شکل ۵-۲: جاسازی ماژول آلتراسونیک در مخزن.



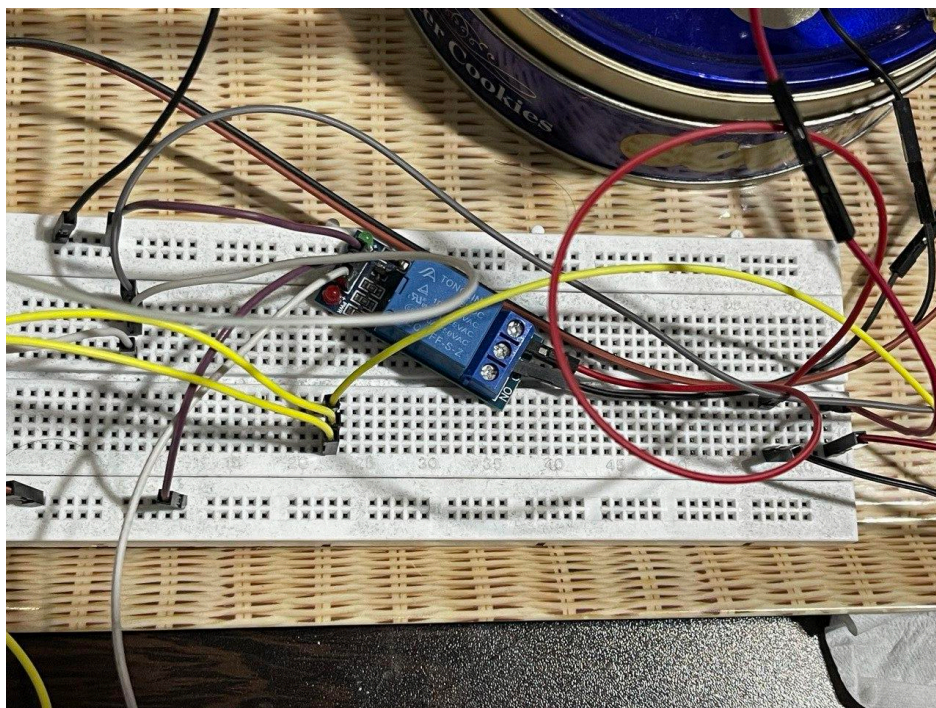
شکل ۵-۳: نمای بالای ماژول آلتراسونیک به همراه قیف ورودی آب.

۳-۵ اتصال ماژول شیربرقی



شکل ۵-۴: اتصال ماژول شیربرقی به مخزن آب.

۴-۵ اتصال رله

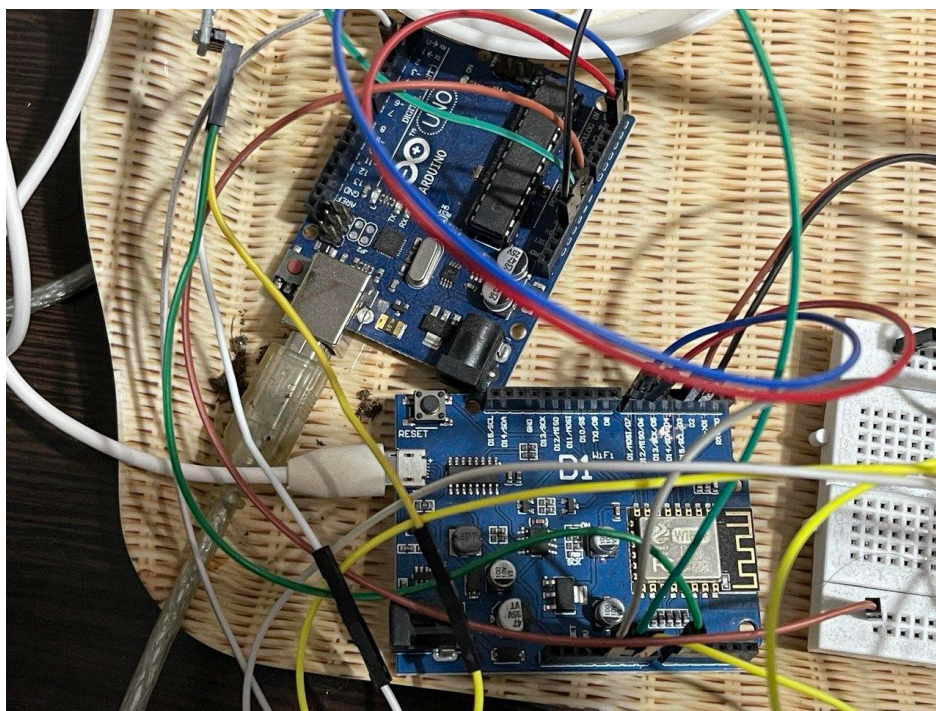


شکل ۵-۵: رله تک کانال بکاربرده شده در مدار.



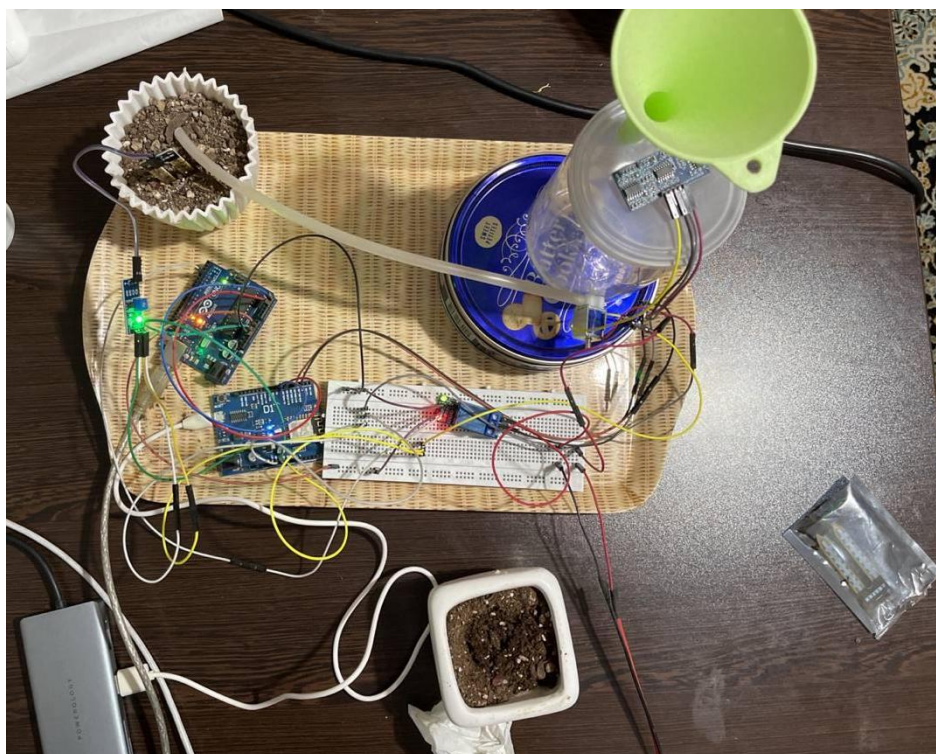
شکل ۵-۶: آداپتور ۱۲ ولتی مورد نیاز برای رله.

۵-۵ اتصال دو برد اصلی به یکدیگر



شکل ۵-۷: اتصال دو برد برای برقراری ارتباط سریال.

۵-۶ جمع‌بندی



شکل ۵-۸: شمای کلی طراحی و مدار پیاده‌سازی شده.

فصل ۶

نحوه پیاده سازی نرم افزار

۶-۱ نحوه اتصال مشتری

```
1 #include <BlynkSimpleEsp8266.h> // Blynk for internet-based connection
2 #include <SoftwareSerial.h>      // Serial Communication package
3
4
5 // Serial Communication port setting
6 SoftwareSerial softSerial(13, 12);
7
8 // States of systems
9 #define AUTO 0
10 #define MANUAL 1
```

۶-۲ تنظیمات اولیه و متغیرها

```
1 // Virtual Ports for Blynk
2 //
3 // V0 -> Manual/ Auto button      //
4 // V1 -> Open/Close of valve button //
5 // V2 -> Threshold of moisture    //
```

```

6 // V3 -> Valve state LED          //
7 // V4 -> Timer Display             //
8 // V5 -> Water level Display       //
9 // V6 -> Soil Moisture Display     //
10 //////////////////////////////////////

1 // You should get Auth Token in the Blynk App.
2 // Go to the Project Settings (nut icon).
3 char auth[] = "G3k5NqlDZoTHM0f8cFcn0VQzkQOMmQ-U";
4 // Your WiFi credentials.
5 char ssid[] = "Mobinnet";
6 char pass[] = "";

```

```

1 // defines variables
2 BlynkTimer timer;
3 float distance;
4 int state = 0;
5 int valve_state = 0;
6 float moist_thr = 500;
7 int soil_moisture = 0;

```

۳-۶ تعیین فاصله ی سطح آب

```

1 float cal_distance() {
2     long duration;
3     //Clear the trigPin
4     digitalWrite(trigPin, LOW);
5     delayMicroseconds(2);
6     // Set the trigPin on HIGH state for 10 micro seconds
7     digitalWrite(trigPin, HIGH);
8     delayMicroseconds(10);
9     digitalWrite(trigPin, LOW);
10    // Read the echoPin, returns the sound wave travel time in
    microseconds

```

```

11     duration = pulseIn(echoPin, HIGH);
12     // Calculating the distance
13     distance = duration * 0.034/2;
14     return distance;
15 }

```

۴-۶ نحوه کار سیستم در هر گردش

```

1 void loop() {
2     Blynk.run();
3     timer.run();
4     distance = (float) ((int) (cal_distance() * 10)) / 10; // Calculate
                    distance to water
5     soil_moisture = analogRead(A0); // read from
                    analog pin A3
6     /// Checking whether we are in auto or manual state.
7     if (state == AUTO) {
8         if (soil_moisture > moist_thr) // Turning valve on, if
                    soil_moisture is above than threshold
9             valve_state = 1;
10        else
11            valve_state = 0;
12    }
13    /// Monitoring values in terminal
14    print_stats();
15    delay(500);
16 }

```

۵-۶ ارتباط با مشتری

۱-۵-۶ فعال سازی سیستم

```

1 void blynk_start_connection () {
2     Blynk.begin(auth, ssid, pass); // Connect to wifi with ssid, pass.
    After that, connect to client with auth.
3     Blynk.connect();
4     timer.setInterval(1000L, update_client); // Here you set interval (1
    sec) and which function to call, We frequently update the state of
    client.
5 }

1 BLYNK_CONNECTED() {
2     Blynk.notify("System turned on");
3     Blynk.syncAll();
4 }

```

۶-۵-۲ دریافت حالت سیستم

```

1 BLYNK_WRITE(V0)
2 {
3     state = param.asInt(); // assigning incoming value from pin V0 to a
    variable
4     Serial.print("Change state to: ");
5     Serial.println(state);
6     if (state == AUTO ) {
7         Blynk.notify("System now on Auto mode!");
8     } else {
9         Blynk.notify("System now on Manual mode!");
10    }
11 }

```

۶-۵-۳ دریافت حالت شیر

```

1 BLYNK_WRITE(V1)
2 {
3     int pinValue = param.asInt(); // assigning incoming value from pin V1
    to a variable

```

```

4   Serial.print("Click on close/open button: ");
5   if (state == AUTO) {
6       Blynk.notify("System on Auto mode, you cannot open/close valve!");
7   };
8   } else {
9       valve_state = pinValue;
10      if (valve_state == 0) {
11          Blynk.notify("Valve is closed.");
12      } else {
13          Blynk.notify("Valve is opened.");
14      }
15  }

```

۴-۵-۶ دریافت مقدار آستان رطوبت

```

1 BLYNK_WRITE(V2)
2 {
3     moist_thr = param.asDouble(); // assigning incoming value from pin V2
4     // to a variable
5     Serial.print("Moisture threshold has been changed to: ");
6     Serial.println(moist_thr);
7 }

```

۵-۵-۶ بروزرسانی وضعیت مشتری

```

1 void update_client() {
2     Blynk.virtualWrite(V5, distance); // Write Water level on blynk
3     Blynk.virtualWrite(V6, soil_moisture); // Write Soil Moisture value
4     // on blynk
5     if(valve_state == 0) {
6         Blynk.virtualWrite(V3, 0); // Turning off LCD in Blynk
7         softSerial.write('0'); // Send a code to Arduino UNO
8         // board
9     }
10 }

```

```

7     } else {
8         Blynk.virtualWrite(V3, 255);           // Turning on LCD in Blynk
9         softSerial.write('1');                 // Send a code to Arduino UNO
10        board
11    }
12 }

```

۶-۶ چاپ متغیرها در ترمینال

```

1 void print_stats () {
2     Serial.print("soil: ");
3     Serial.println(soil_moisture);
4     Serial.print("Distance: ");
5     Serial.print(distance);
6     Serial.print(" Valve_state: ");
7     Serial.println(valve_state);
8 }

```

۶-۷ نحوه کار برد آردینو UNO

۶-۷-۱ ارسال کد از طرف مدار اصلی

```

1 BLYNK_WRITE(V1)
2 {
3     int pinValue = param.asInt(); // assigning incoming value from pin V1
4     to a variable
5     Serial.print("Click on close/open button: ");
6     if (state == AUTO) {
7         Blynk.notify("System on Auto mode, you cannot open/close valve!");
8     };
9     } else {
10        valve_state = pinValue;
11        if (valve_state == 0) {

```



```
10         Blynk.notify("Valve is closed.");
11     } else {
12         Blynk.notify("Valve is opened.");
13     }
14 }
15 }
```

۶-۷-۲ نحوه کار در هر گردش

```
1 void loop()
2 {
3     if (softSerial.available()) { // If signal from main board available
4         val = softSerial.read();
5         if(val == OFF) {
6             digitalWrite(relay_port, LOW);    // Turn off relay
7         } else if (val == ON) {
8             digitalWrite(relay_port, HIGH);    // Turn on relay
9         }
10    }
11    delay(200);
12 }
```

فصل ۷

جمع بندی

۱- مطالب تکمیلی

مراجع

واژه‌نامه

الف

ب

پ

ت

ج

چ

ح

خ

د

ر

ز

س