

UNIVERSIDADE FEDERAL DE VIÇOSA
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

DANIEL SOSSAI HUPSEL CELESTINO

**DESENVOLVIMENTO DE UM SISTEMA DE IRRIGAÇÃO
AUTOMATIZADO PARA CAFEZAIS UTILIZANDO A
PLATAFORMA ARDUINO**

VIÇOSA
2023

DANIEL SOSSAI HUPSEL CELESTINO

**DESENVOLVIMENTO DE UM SISTEMA DE IRRIGAÇÃO
AUTOMATIZADO PARA CAFEZAIS UTILIZANDO A
PLATAFORMA ARDUINO**

Monografia apresentada ao Departamento de Engenharia Elétrica do Centro de Ciências Exatas e Tecnológicas da Universidade Federal de Viçosa, para a obtenção dos créditos da disciplina ELT 402 – Projeto de Engenharia II – e cumprimento do requisito parcial para obtenção do grau de Bacharel em Engenharia Elétrica.

Orientador: Prof. Dr. Denilson Eduardo Rodrigues

VIÇOSA
2023

DANIEL SOSSAI HUPSEL CELESTINO

**DESENVOLVIMENTO DE UM SISTEMA DE IRRIGAÇÃO
AUTOMATIZADO PARA CAFEZAIS UTILIZANDO A PLATAFORMA
ARDUINO**

Monografia apresentada ao Departamento de Engenharia Elétrica do Centro de Ciências Exatas e Tecnológicas da Universidade Federal de Viçosa, para a obtenção dos créditos da disciplina ELT 402 – Projeto de Engenharia II – e cumprimento do requisito parcial para obtenção do grau de Bacharel em Engenharia Elétrica.

Aprovada em 23 de novembro de 2023.

COMISSÃO EXAMINADORA



Documento assinado digitalmente

DENILSON EDUARDO RODRIGUES

Data: 30/11/2023 16:15:29-0300

Verifique em <https://validar.iti.gov.br>

Prof. Dr. Denilson Eduardo Rodrigues – Orientador
Universidade Federal de Viçosa



Documento assinado digitalmente

MAURO DE OLIVEIRA PRATES

Data: 24/11/2023 09:15:48-0300

Verifique em <https://validar.iti.gov.br>

Prof. Dr. Mauro de Oliveira Prates – Primeiro Avaliador
Universidade Federal de Viçosa



Documento assinado digitalmente

LUCAS JONYS RIBEIRO SILVA

Data: 24/11/2023 08:46:41-0300

Verifique em <https://validar.iti.gov.br>

Me. Lucas Jonys Ribeiro Silva – Segundo Avaliador
Universidade de São Paulo

*“Ninguém sabe tudo. Ninguém ignora tudo. Por isso aprendemos sempre”
(Paulo Freire)*

Agradecimentos

Agradeço primeiramente aos meus amados pais, Luis Carlos e Eloizia, e irmã, Camila, por serem uma fonte inesgotável de amor, apoio e incentivo ao longo dessa jornada. Aos meus tios Luiz e Myriam por todo suporte no início da minha trajetória acadêmica, vocês fazem parte dessa conquista.

Aos meus avós, Albino, Jurandy (*in memorian*), Antônio e Edelzy pelo incentivo aos estudos.

Aos meus amigos Ricardo Ferreira, Danilo Cunha, Gabriel Rabelo, Gabriell Hott, Lucas Timo, Felipe Faccioli, Alexandre Pires, Lucas Paiva e Guilherme Barreto por toda parceria, esses anos não teriam sido os mesmos sem vocês.

Aos amigos da Lateria, Atlético das Engenharias, Banda Lateria e Pagoteria pelos momentos de descontração e alegrias compartilhadas.

Aos amigos da ELT15 que dividiram praticamente todos os dias dessa trajetória comigo, tenham a certeza que o caminho foi muito mais fácil e mais leve com a presença de vocês.

Aos funcionários e professores do Departamento de Engenharia Elétrica da UFV, em especial ao professor Denilson Eduardo Rodrigues pela orientação nesse projeto de conclusão de curso e a todos que de alguma forma me apoiaram nesses anos em Viçosa.

Resumo

Um dos grandes problemas da agricultura, de modo geral, é a escassez hídrica devido à distribuição desuniforme das chuvas durante o ano. Atualmente os produtores rurais tem optado por variedades mais produtivas ante a variedades mais resistentes à seca, pois o recurso hídrico necessário para produção pode ser suprido com sistemas de irrigação.

O objetivo deste trabalho foi criar um sistema de irrigação automatizado para reduzir a mão de obra e custo de energia em uma lavoura de Café Conilon, localizada no norte do Espírito Santo. Foi utilizada a plataforma Arduino para processar sinais enviados por sensores de umidade do solo e de chuva e enviar dados para a abertura e fechamento de quatro válvulas solenóide.

O início do ciclo de irrigação é definido pelo percentual de umidade do solo, mas o seu término é definido unicamente pelo tempo duração do ciclo, uma vez que os resultados ficam comprometidos em caso de alteração na profundidade de inserção do sensor e tipo do solo.

Este protótipo foi criado de acordo com as particularidades encontradas em campo, onde a irrigação é feita por gravidade e por conta da pressão da água necessária no interior da tubulação para o bom funcionamento da mesma, apenas um setor é irrigado por vez.

O protótipo comportou-se da maneira esperada no que diz respeito à abertura e fechamento das válvulas, respeitando as condições impostas na programação do microcontrolador. Contudo, duas ressalvas merecem destaque: em relação ao tipo de sensor de umidade utilizado, nota-se a necessidade de utilizar sensores mais robustos e precisos; e quanto ao microcontrolador, foram identificados pontos de travamento durante chaveamentos muito rápidos após um longo período em funcionamento, o que demonstra a necessidade de um processamento mais rápido.

Abstract

One of the major challenges in agriculture, in general, is water scarcity due to the uneven distribution of rainfall throughout the year. Currently, farmers have been opting for more productive varieties over drought-resistant ones, as the necessary water resources for production can be supplied through irrigation systems.

The objective of this work was to create an automated irrigation system to reduce labor and energy costs in a Conilon Coffee plantation located in the northern region of Espírito Santo. The Arduino platform was used to process signals sent by soil moisture and rain sensors and to transmit data for the opening and closing of four solenoid valves.

The initiation of the irrigation cycle is defined by the percentage of soil moisture, but its termination is solely determined by the cycle duration time, as the results are compromised in case of changes in sensor insertion depth and soil type.

This prototype was developed in accordance with the specific conditions found in the field, where irrigation is gravity-based. Due to the required water pressure within the pipeline for its proper functioning, only one sector is irrigated at a time.

The prototype behaved as expected regarding the opening and closing of the valves, adhering to the conditions imposed by the microcontroller programming. However, two caveats deserve attention: regarding the type of soil moisture sensor used, there is a discernible need for more robust and precise sensors; and concerning the microcontroller, points of lockup were identified during very rapid switchings after a long operational period, indicating the need for faster processing.

Sumário

1	Introdução.....	9
2	Hipótese.....	11
3	Impactos do Projeto.....	12
4	Objetivos	14
5	Referencial Teórico	15
5.1	Sistemas de Irrigação Automatizados.....	15
5.2	Sistemas de irrigação com unidades temporizadas.....	15
5.3	Sistemas de irrigação sem unidades temporizadas	15
5.4	Sistemas com sensor de umidade.....	16
6	Materiais e Métodos	17
6.1	Software	17
6.2	Materiais	18
6.2.1	Arduino Mega 2560.....	18
6.2.2	Módulo RTC DS3231.....	19
6.2.3	Sensor de Temperatura e Umidade DHT-22 AM2302.....	19
6.2.4	Válvula Solenoide NF	20
6.2.5	Sensor de Umidade do Solo Higrômetro.....	20
6.2.6	Módulo Sensor de Chuva YL-83 9SS18	21
6.2.7	Display LCD 16x2 Backlight Azul	22
6.2.8	Módulo Relé 5V	22
6.3	Métodos	23
7	Resultados	25
8	Conclusão	29
9	Referências.....	30

Lista de Figuras

Figura 1 - Método de desvio de curso d'agua para irrigação de plantios	9
Figura 2 - Aspersores de Irrigação em atuação	12
Figura 3 - Interface Arduino IDE	17
Figura 4 - Interface do Fritzing	18
Figura 5 - Arduino MEGA 2560 (ARDUINO)	18
Figura 6 - Modulo RTC DS3231 (FILIPEFLOP)	19
Figura 7 - Sensor de umidade e temperatura DHT-22 AM2302 (FILIPEFLOP).....	20
Figura 8 - Figura 8 - Válvula Solenóide 180° 1/2" (ELETROGATE).....	20
Figura 9 - Sensor de Umidade do Solo Higrômetro (SMARTKITS).....	21
Figura 10 - Sensor de Chuva YL-83 9SS18 (FILIPEFLOP).....	21
Figura 11 - Display LCD 16x2 Backlight Azul (FILIPEFLOP)	22
Figura 12 - Módulo Relé 5V.....	22
Figura 13 - Esquemático do protótipo	24
Figura 14 - LCD indicando o funcionamento do Setor 4	25
Figura 15 – Representação visual do protótipo em funcionamento	25
Figura 16 - Sistema de tubulação e válvulas utilizado	26
Figura 17 - Bancada de testes do protótipo	26
Figura 18 - Visão do monitor serial indicando chuva forte.....	27

1 *Introdução*

Desde os primórdios da civilização as técnicas de irrigação já eram utilizadas pelo homem em seus cultivos. Foi observado que as plantas se tornavam mais viçosas e exuberantes nos períodos mais úmidos do ano. Através do desvio de cursos d'água (Figura 1) deu-se início à forma mais rudimentar da irrigação, levando prosperidade no cultivo e melhora da qualidade de vida da população de regiões áridas e semiáridas. (TESTEZLAF, 2017)

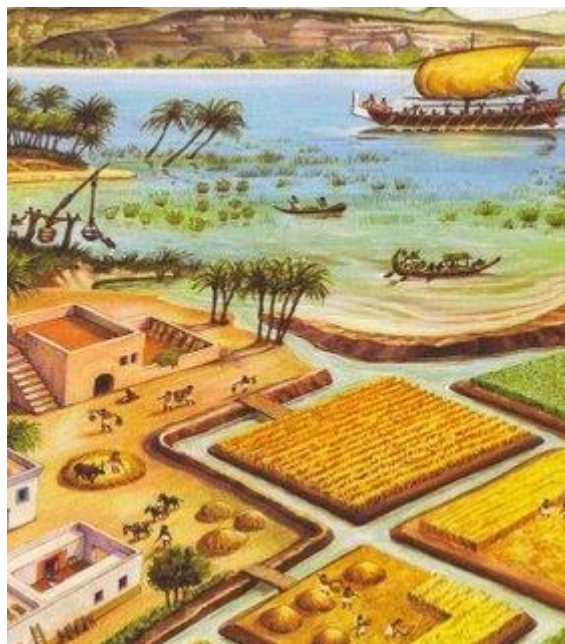


Figura 1 - Método de desvio de curso d'água para irrigação de plantios

Embora a história da agricultura no Brasil mostre que as áreas de cultivo estavam localizadas em regiões úmidas, há registros de que no ano de 1589 os jesuítas já praticavam a técnica da irrigação na Fazenda Santa Cruz, no estado do Rio de Janeiro. Em cafezais, a técnica se iniciou na década de 1950 com a utilização da aspersão no estado de São Paulo. (TESTEZLAF, 2017)

A utilização da irrigação na cafeicultura redesenhou a distribuição geográfica do cultivo de café no Brasil, incorporando áreas antes não recomendadas para o plantio e transformando-as em novos polos de desenvolvimento da cultura e das regiões. (VICENTE et al., 2015).

Estima-se que a área irrigada de cafeeiro Conilon no Espírito Santo seja de 150 mil hectares sendo a grande maioria na região norte capixaba (Teixeira et al., 2012). Segundo dados da Conab (2016), o Espírito Santo possui 286.371 hectares plantados com cafeeiro, sendo assim, 52,4% da cafeicultura no estado é irrigada.

Atualmente muito tem se falado a respeito do uso racional da irrigação devido escassez da água e necessidade de economia, uma vez que muitos produtores rurais utilizam tal recurso de forma desenfreada, por vezes até fazendo mal ao cultivo, haja vista que o excesso de água também é prejudicial à cultura do cafeeiro. (H.R.O. SILVEIRA). Faz-se então necessário um método para controle da quantidade de água disponível à planta.

O uso de um sistema de irrigação automatizado busca não só a economia de água, mas também economizar mão de obra e energia elétrica, dado que a motobomba pode ser programada para entrar em funcionamento apenas em horários com tarifas de energia reduzidas, além de evitar o acionamento da mesma quando o solo ainda estiver úmido.

Neste trabalho é apresentado um protótipo de irrigação automatizada para cafeeiros utilizando a plataforma Arduino.

2 Hipótese

Em resposta ao problema do uso desenfreado da água e consumo energético alto das irrigações, considera-se a hipótese de que o controle de umidade do solo garantirá a entrega da quantidade necessária para o desenvolvimento adequado do café, não deixando o solo com carência nem excesso de água, não havendo desperdício, e assim a planta se desenvolverá de forma saudável, podendo aumentar sua produção.

Também pode se considerar a hipótese do conseqüente benefício da redução no valor da tarifa de energia elétrica, uma vez que a bomba de água para irrigação não será ligada sem a devida necessidade, e por mais tempo que o necessário.

Sendo assim, o consumo de recursos hídricos também será feito de maneira responsável, utilizando somente o necessário e economizando água.

O desenvolvimento de um projeto de sistema de irrigação automatizado utilizando a plataforma Arduino, é muito útil, considerando o baixo custo, a facilidade de utilização, sendo acessível a pequenos e grandes produtores rurais.

3 *Impactos do Projeto*

Cada vez mais a tecnologia tem ganhado espaço no meio rural, os agricultores estão investindo em máquinas e aplicando novas tecnologias em suas lavouras, para evoluir e obter bons resultados, dessa forma melhorando e ampliando suas plantações, tendo qualidade, produtividade e diminuindo a mão de obra.

Caracterizada como agricultura de precisão, essa implementação tecnológica no campo, com maior enfoque em uso de tecnologia e máquinas, propõe o uso de várias tecnologias, visando reduzir os custos dos processos de produção agrícola.

A irrigação (Figura 2) é um meio que proporciona aos produtores rurais, mais expectativa de uma boa safra, independente das chuvas na região. Podendo assim, diminuir os riscos de perda de safra, e de gastos com investimentos que não vão surtir efeito com a falta de chuvas.

O projeto tem como objetivo a criação de um sistema de irrigação automatizado para aplicação em propriedades rurais, sendo de baixo custo e fácil manutenção, o que facilita o seu uso. Desse modo, podendo diminuir o desgaste de tempo útil humano para monitoramento e execução da irrigação, podendo este tempo ser direcionado a outras tarefas complexas requeridas por uma área de agricultura.



Figura 2 - Aspersores de Irrigação em atuação

Além disso, o projeto tem por objetivo o consumo consciente da água, reduzindo o desperdício, só irrigando o solo em dias necessários, de forma automatizada com carga

hídrica definida, e com precisão. Em dias de chuvosos, por exemplo, o protótipo irá realizar um novo ciclo de irrigação após o sensor de umidade detectar a necessidade, por meio de monitoramento da umidade do solo, bem como a umidade relativa do ar. O uso da água se dá pela utilização do sensor de umidade, que realiza medições, garantindo que o solo está recebendo água suficiente para o desenvolvimento da lavoura

Portanto, o objetivo principal desse protótipo é tornar acessível, de forma automatizada, o recurso hídrico para a irrigação da plantação sem a necessidade de intervenção humana, e aliado a isso, maximizar a utilização do recurso hídrico, sendo um projeto economicamente viável e ambientalmente responsável.

4 *Objetivos*

Este trabalho tem como objetivo geral desenvolver um protótipo de um sistema de irrigação automatizado para uma lavoura de café Conilon localizada no norte do Espírito Santo, utilizando a plataforma de prototipagem Arduino MEGA2560.

Dado o objetivo geral, têm-se como objetivos específicos:

- Realizar um levantamento de sensores essenciais para determinação de necessidade diária de irrigação do plantio, bem como a instalação de válvulas solenoides nos irrigadores presentes;
- Desenvolver um código no Arduino IDE que torne o Arduino capaz de realizar as leituras dos sensores e fazer o controle das válvulas solenoides de forma automatizada;
- Estudar a viabilidade de implantação do protótipo em um mundo real.

5 Referencial Teórico

5.1 Sistemas de Irrigação Automatizados

Segundo Reis (REIS, 2015), há uma grande discussão sobre as diferentes tecnologias de sistemas de irrigação automatizados, devido as melhorias na qualidade do processo de cultivo, pois possibilitam irrigações sem a necessidade de acompanhamento humano, além de otimizar o consumo de água. Para Durson (DURSON & OZDEN, 2011) devido ao crescimento da demanda de água doce faz-se necessário o uso da tecnologia de automação para a melhor utilização dos recursos hídricos.

Atualmente os sistemas automatizados de irrigação se dividem em dois grupos principais dado o seu princípio de funcionamento: baseados em unidades temporizadas e unidades sem temporização.

5.2 Sistemas de irrigação com unidades temporizadas

Segundo Zazueta (ZAZUETA, 1994), estes sistemas de irrigação realizam a liberação de água em intervalos fixos de tempo. Entretanto, este tipo de sistema ocasiona um desperdício de água por conta da liberação de água quando o solo ainda se encontra úmido, além de não se preocupar com a temperatura do ambiente ao realizar a irrigação podendo ocorrer desperdício de água pela evapotranspiração ou lixiviação, dentre outros problemas.

5.3 Sistemas de irrigação sem unidades temporizadas

Esse tipo de sistema é baseado em um sistema de controle alimentado por sensores que identificam o momento ideal de irrigar, bem como a quantidade de água necessária.

Nise (NISE, 2011) afirma que os sistemas de controle podem compensar a falta de habilidade humana e mecanismos projetados sobre eles podem ser usados para substituir humanos nos trabalhos em locais remotos, perigosos ou propiciar conveniência.

A utilização de um sistema de controle em um sistema de irrigação tem como objetivo é automatizar a ação manual de irrigar as plantas e tentar compensar as

perturbações provocadas por variações climáticas, tipos de solos, tipos de plantas e variações na temperatura (NISE, 2011).

5.4 Sistemas com sensor de umidade

Segundo Durson (DURSON & OZDEN, 2011), a aplicação de uma rede de sensores permite fazer o monitoramento em tempo real da quantidade de água presente no solo, determinando onde, quando e quanto irrigar, proporcionando um uso eficiente de recursos e eliminando a necessidade de mão-de-obra para efetuar a irrigação.

Reis (REIS, 2015) ainda afirma que o sucesso das atividades agrícolas está relacionado ao uso eficiente da água nas culturas principalmente em períodos de estresse hídrico e isso é atingido fazendo-se o controle da umidade do solo.

6 *Materiais e Métodos*

6.1 Software

Arduino IDE

O Arduino IDE (*Integrated Development Environment*, em português, Ambiente de Desenvolvimento Integrado) (Figura 3) é um software multiplataforma, escrito na linguagem JAVA, utilizado para escrever e enviar programas para a placa Arduino, no qual suporta as linguagens C e C++ (ARDUINO).

A screenshot of the Arduino IDE interface. The window title is "Blink | Arduino 1.8.5". The main editor area shows the following code:

```
This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}
```

The status bar at the bottom indicates "32" and "Arduino/Genuino Uno on COM1".

Figura 3 - Interface Arduino IDE

Fritzing

O Fritzing (Figura 4) foi desenvolvido na Fachhochschule Potsdam (University of Applied Sciences Potsdam), na Alemanha, e lançado em sua primeira versão 0.1b em novembro de 2008. Trata-se de um software de código aberto, sob licença GPLv3, para desenvolvimento de protótipos de circuitos eletrônicos. Este software permite que sejam criados layouts de circuitos eletrônicos de maneira simples e rápida que permitem a documentação dos esquemáticos por pesquisadores, designers, artistas e hobbistas. (Kenshima, 2023)

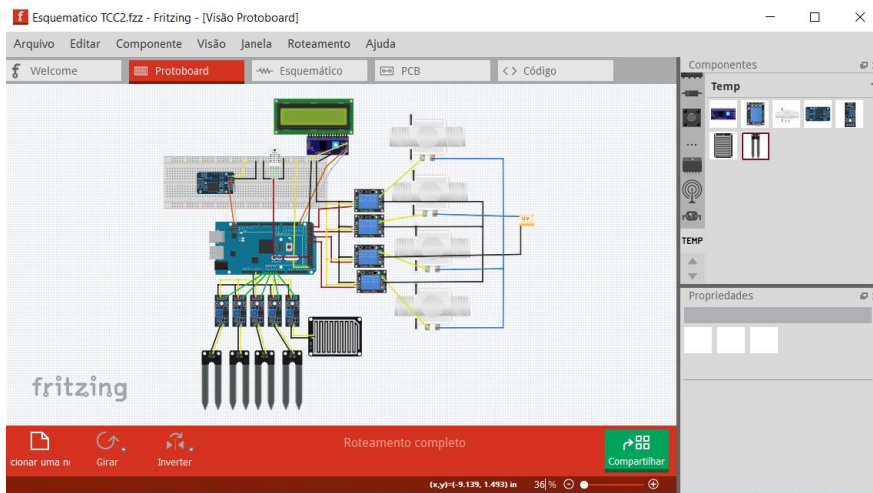


Figura 4 - Interface do Fritzing

6.2 Materiais

Arduino Mega 2560

O Arduino Mega 2560 (Figura 5) é uma placa microcontrolada da plataforma Arduino, baseada no ATmega2560. Possui 54 pinos de entrada/saída digital (dos quais 15 podem ser usados como saídas PWM), 16 entradas analógicas, 4 UARTs (portas seriais de hardware), um oscilador de cristal de 16 MHz, uma conexão USB, um conector de alimentação, um conector ICSP, e um botão de reset. Esta placa contém tudo o que é necessário para dar suporte ao microcontrolador, basta conectá-lo a um computador com um cabo USB ou ligá-lo com um adaptador AC-DC ou bateria para começar. (ARDUINO)



Figura 5 - Arduino MEGA 2560 (ARDUINO)

Módulo RTC DS3231

O Real Time Clock (RTC) DS3231 (Figura 6) é um relógio de tempo real de alta precisão e baixo consumo de energia, com protocolo I2C. Em sua placa vem embutido um sensor de temperatura e um cristal oscilador para melhor exatidão. Este módulo é capaz de fornecer informações como segundo, minuto, dia, mês e ano. Podendo operar no formato 12 horas ou 24 horas, possui correções automáticas para meses com menos de 31 dias e anos bissextos. Em caso de detecção de falta de energia, o módulo aciona automaticamente a bateria para evitar a perda de dados.



Figura 6 - Módulo RTC DS3231 (FILIPEFLOP)

Sensor de Temperatura e Umidade DHT-22 AM2302

O DHT-22 AM2302 (Figura 7) é um sensor de temperatura e umidade com interface serial a um fio (single wire). O sensor é calibrado de fábrica e não necessita de hardware adicional para funcionar. Possui amplitude de medição 120 graus Celsius, podendo medir entre -40 a 80 graus Celsius e umidade entre 0 e 100%. Sua arquitetura com apenas um pino digital de comunicação serial proporciona uma facilidade maior de uso no Arduíno quando comparado com outros sensores. (BAÚ DA ELETRÔNICA)



Figura 7 - Sensor de umidade e temperatura DHT-22 AM2302 (FILIFELOP)

Válvula Solenoide NF

A Válvula Solenóide Normalmente Fechada (Figura 8) é usada para controlar o fluxo de água. Quando alimentada com 12V DC, ela abre e permite a passagem de fluxo de água. Ao desenergizá-la, ela fecha e corta o fluxo. Seu princípio de funcionamento é baseado em uma bobina (solenóide), que quando energizada induz um campo eletromagnético responsável por movimentar o êmbolo da válvula. (ELETROGATE)



Figura 8 - Válvula Solenóide 180° 1/2" (ELETROGATE)

Sensor de Umidade do Solo Higrômetro

O Sensor de Umidade do Solo Higrômetro (Figura 9) é utilizado para detectar as variações de umidade do solo. É um sensor digital, logo, o valor de saída é nível lógico alto ou baixo. A saída do sensor fica em nível lógico alto quando a umidade do solo estiver abaixo de um valor pré ajustado via

potenciômetro e a saída fica em nível lógico baixo quando a umidade do solo estiver acima do valor ajustado. Possui um comparador LM393, LED indicador para presença de tensão e LED indicador para saída digital. (SMARTKITS)

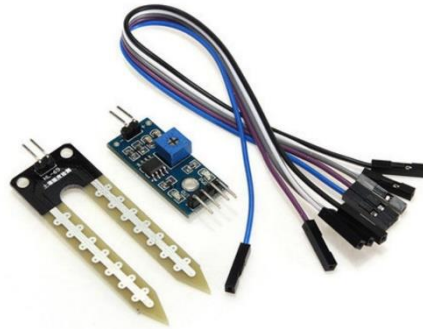


Figura 8 - Sensor de Umidade do Solo Higrômetro (SMARTKITS)

Módulo Sensor de Chuva YL-83 9SS18

O Sensor de Chuva YL-83 9SS18 (Figura 10) é do tipo resistivo podendo monitorar condições climáticas como chuva e neve. Quando o clima está seco, a saída do sensor fica em nível lógico alto e quando há uma gota de chuva, a saída fica em estado baixo. O ajuste limite entre tempo seco e chuva pode ser realizado através de um potenciômetro presente no sensor que regulará a saída digital D0. Possui LED indicador de tensão, LED indicador para saída digital e opera em tensões entre 3,5 e 5 V. (ELETROGATE)



Figura 9 - Sensor de Chuva YL-83 9SS18 (FILIPEFLOP)

Display LCD 16x2 Backlight Azul

Este display possui o controlador HD44780 (Figura 11) utilizado em toda a indústria de LCD's como base de interface. São 16 colunas por 2 linhas, backlight azul e escrita branca. É um display simples, econômico, mas útil para o que se propõe. Opera com tensões entre 4,5 a 5,5 V e corrente entre 1 e 1,5 mA. (FILIPEFLOP)



Figura 10 - Display LCD 16x2 Backlight Azul (FILIPEFLOP)

Módulo Relé 5V

O módulo Relé de 5V (Figura 12) permite uma integração com uma ampla gama de microcontroladores como Arduino, PIC, AVR, entre outros. A partir das saídas digitais, pode-se controlar cargas maiores como motores AC, eletroímãs e solenóides. O módulo é equipado com um relé de alta qualidade com carga nominal que pode chegar a 10 A e 250 VAC. (ELETROGATE)



Figura 11 - Módulo Relé 5V

6.3 Métodos

Para este estudo, uma bancada foi montada com quatro vasos de plantas, simulando 4 setores do sistema de irrigação (cada setor é constituído de aproximadamente 1500 plantas de café), sendo assim o vaso de planta é o objeto a ter a irrigação controlada.

Cada vaso de planta possuía um sensor de umidade, uma válvula solenóide e era irrigado por meio de um microaspersor do tipo bailarina, com a água sendo levada até ele por meio de uma mangueira de ½ polegada e tendo origem em uma torneira residencial. O sensor de chuva foi posicionado em lugar desabrigado, a uma distância de aproximadamente 30cm de cada vaso de planta e o módulo de umidade e temperatura do ar, estava acoplado ao protótipo.

Por meio do Arduino MEGA foi feita uma varredura das variáveis dos sensores que poderão ser lidas a qualquer momento pelo usuário, além de realizar o acionamento do sistema de irrigação se as condições pré-definidas no software forem satisfeitas.

O ciclo de irrigação foi definido para ter início quando a umidade do solo estiver abaixo de 40% e será interrompido após 10 minutos do início. O seu funcionamento foi pré-definido para funcionar apenas durante a noite, com o intuito de economizar energia. Caso o sensor de chuva indique 20% de umidade, o ciclo será impedido de começar.

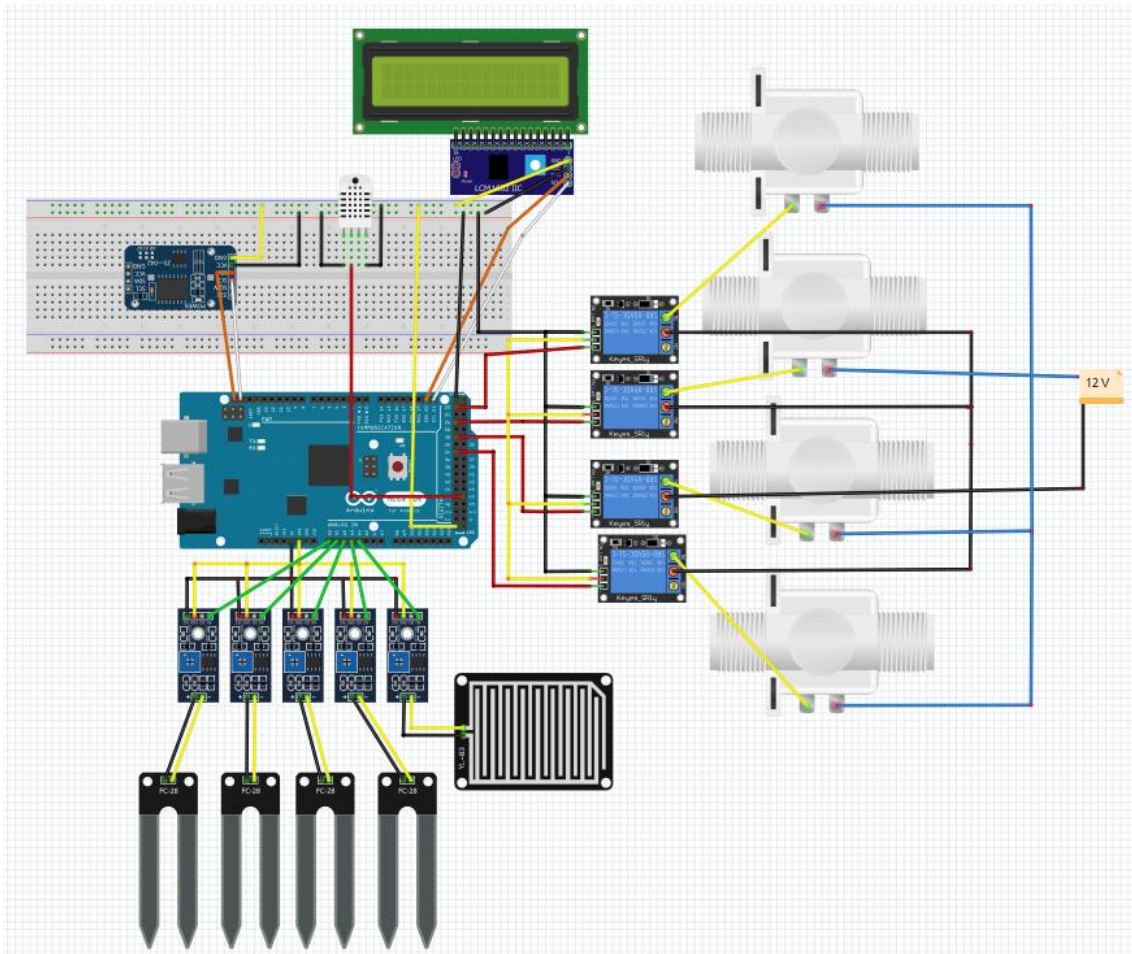


Figura 12 - Esquemático do protótipo

7 *Resultados*

Nesta seção, são apresentados os resultados obtidos com este trabalho e possíveis pontos de melhorias identificados.

Nas Figuras de 14 a 17 é mostrada a estrutura do protótipo.

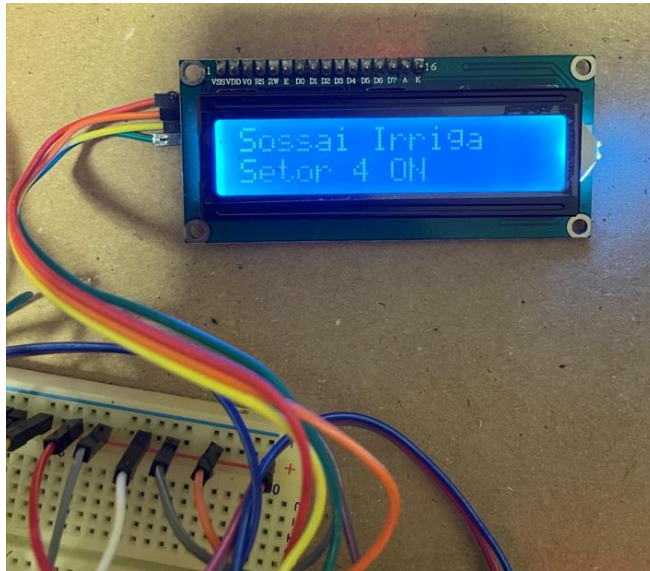


Figura 13 - LCD indicando o funcionamento do Setor 4



Figura 14 – Representação visual do protótipo em funcionamento

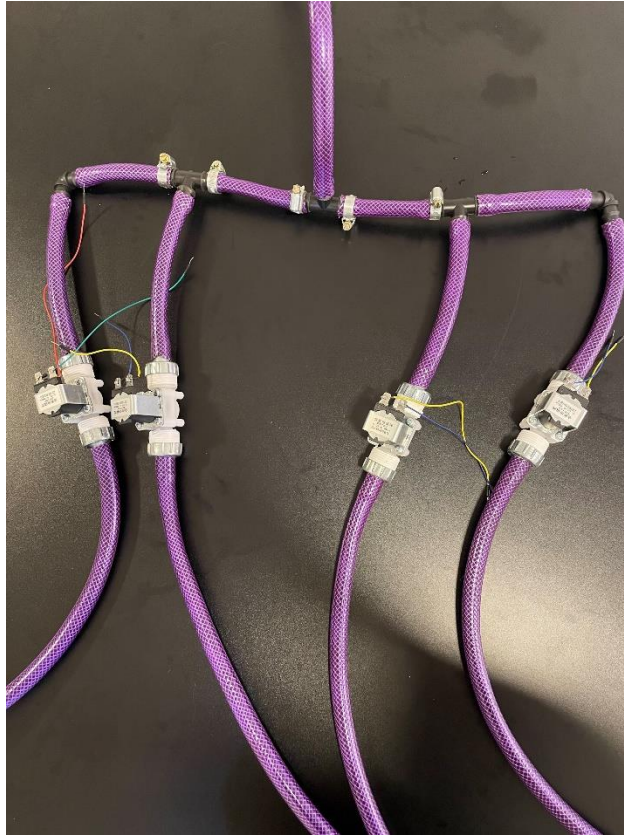


Figura 15 - Sistema de tubulação e válvulas utilizado



Figura 16 - Bancada de testes do protótipo

Os resultados obtidos em bancada indicam que o protótipo se mostrou eficiente em relação aos objetivos propostos, pois foi capaz de realizar o acionamento das válvulas solenóides pelo tempo de 10 minutos cada e seguindo os critérios de acionamento estabelecidos na programação do microcontrolador.

Durante os testes notou-se uma grande variação nas medições dos sensores de umidade, onde para um mesmo vaso de plantas, com uma distância de 2 cm entre cada sensor, os valores obtidos apontavam até 16% de variação, conforme mostrado na Tabela 1 abaixo.

Tabela 1 - Resultados dos testes de umidade

	Teste 1	Teste 2	Teste 3
Sensor 1	52.41	81.52	35.12
Sensor 2	57.17	74.63	42.89
Sensor 3	47.69	77.22	41.98
Sensor 4	41.44	82.76	36.84

Essa alteração entre as medições de um mesmo setor corrobora com a hipótese apresentada anteriormente de que a profundidade de inserção do sensor de umidade ou tipo de solo interfere na calibração do protótipo, sendo a duração do ciclo determinado pelo tempo de irrigação o melhor caminho para se obter êxito na irrigação da lavoura.

```
----- FIM -----  
  
Sossai Irriga  
Data: 24/6/2023  
Hora: 14:32:34  
Dia da semana: Sabado  
Temperatura: 22.90 °C  
Umidade do ar: 65.60%  
Chuva: Forte  
Umidade do solo no setor 1: 21.80  
Umidade do solo no setor 2: 66.57  
Umidade do solo no setor 3: 59.14  
Umidade do solo no setor 4: 74.29  
----- FIM -----
```

Figura 17 - Visão do monitor serial indicando chuva forte

Através do monitor serial, como mostra a Figura 17, é possível acompanhar a cada 10 segundos os níveis de umidade em cada setor, a representação da medição da temperatura e umidade do ar obtidas pelo sensor DHT22, as representações de data, hora e dia da semana obtidas através do RTC e a intensidade da chuva obtida pelo YL-83.

Foi observado também que após um longo tempo em funcionamento, para chaveamentos muito rápidos, em que dois ou mais sensores apontam necessidade de irrigar, o sistema apresentou um leve travamento que não comprometeu o tempo de irrigação, mas que demonstra que para sistemas mais robustos é necessário um microcontrolador com maior capacidade de processamento.

8 *Conclusão*

A irrigação automatizada tem se mostrado uma solução promissora para otimizar o processo de irrigação em diversos setores como agricultura, jardins e até na pecuária. A integração do Arduino, um microcontrolador versátil e de baixo custo, com os mais diversos sensores e atuadores permitiu o desenvolvimento de um sistema inteligente e eficiente de irrigação.

Ao longo dos estudos e experimentos realizados, observou-se que a irrigação automatizada oferece uma série de benefícios. Em primeiro lugar, proporciona uma maior precisão e controle sobre a quantidade de água aplicada, evitando desperdícios e garantindo um uso mais eficiente desse recurso tão valioso para a agricultura. Além disso, o monitoramento contínuo da umidade do solo por meio dos sensores, permite inibir o estado de estresse hídrico provocado pela ausência de água nas raízes, bem como a super irrigação, que eventualmente pode causar doenças, apodrecimento do sistema radicular e lixiviação do solo.

Para melhoria do protótipo em projetos futuros, haja vista o desejo de se aplicar em escala real, é recomendável o desenvolvimento de um sistema de potência para acionamento de bombas centrífugas acima de 5 CV, criação de um sistema de acionamento remoto podendo ser a rádio ou internet (TCP/IP) e uma interface homem-máquina (IHM) para uma melhor interação e controle do processo de irrigação.

9 Referências

- Arduino. (s.d.). *Arduino Uno Rev3*. Acesso em 10 de 11 de 2022, disponível em Arduino: <https://store.arduino.cc/usa/arduino-uno-rev3>
- Arduino. (s.d.). *Arduino Uno Rev3*. (© 2019 Arduino) Acesso em 10 de 11 de 2022, disponível em Arduino: <https://store.arduino.cc/usa/arduino-uno-rev3>
- Baú da Eletrônica. (s.d.). *Ethernet Shield W5100 para Arduino*. Acesso em 10 de 11 de 2022, disponível em Baú da Eletrônica: <https://www.baudaeletronica.com.br/ethernet-shield-w5100-para-arduino.html>
- Baú da Eletrônica. (s.d.). *Módulo Sensor de umidade e temperatura DHT22*. Acesso em 10 de 11 de 2022, disponível em Baú da Eletrônica: <https://www.baudaeletronica.com.br/produto/modulo-sensor-de-umidade-e-temperatura-dht22.html>
- C. Dorf, R., & H. Bishop, R. (2001). *Sistemas de Controle Moderno* (8a ed.). Rio de Janeiro: LTC.
- E. F. W. Lima, J. (s.d.). *O uso da Irrigação no Brasil*. EMBRAPA.
- Eletrogate. (s.d.). *Válvula Solenóide de Entrada de Água - 180° 1/2" x 1/2" 12V DC*. Acesso em 10 de 06 de 2023, disponível em Eletrogate: <https://www.eletrogate.com/valvula-solenoid-de-entrada-de-agua-180-1-2-12v-dc>
- Filipeflop C.E. (s.d.). *Módulo Relé 5V 2 Canais*. Acesso em 10 de 11 de 2022, disponível em Filipeflop: <https://www.filipeflop.com/produto/modulo-rele-5v-2-canal/>
- Filipeflop C.E. (s.d.). *Sensor de Fluxo de Água 1/2" YF-S201b*. Acesso em 10 de 11 de 2022, disponível em Filipeflop: <https://www.filipeflop.com/produto/sensor-de-fluxo-de-agua-12-yf-s201/>
- Filipeflop C.E. (s.d.). *Sensor de Umidade do Solo Higrômetro*. Acesso em 10 de 11 de 2022, disponível em Filipeflop: <https://www.filipeflop.com/produto/sensor-de-umidade-do-solo-higrometro/>
- Filipeflop C.E. (s.d.). *Sensor de Umidade e Temperatura AM2302 DHT22*. Acesso em 10 de 11 de 2022, disponível em Filipeflop: <https://www.filipeflop.com/produto/sensor-de-umidade-e-temperatura-am2302-dht22/>
- Kenshima, G. (s.d.). *Fritzing: Aprenda o que é e como usar de forma simples!* (FilipeFlop Componentes Eletrônicos) Acesso em 10 de 06 de 2023, disponível em <https://www.filipeflop.com/blog/introducao-ao-fritzing/>
- K-Technics Solutions. (s.d.). *K-Technics*. Acesso em 20 de 11 de 2012, disponível em DS3231N AT24C32 IIC RTC Module Precision Clock Module for Arduino:

<https://ktechnics.com/shop/arduino-shields-modules/ds3231-at24c32-iic-precision-real-time-clock-module-memory-module-arduino-3/>

Silva Souza, R. (s.d.). *Curiosidade: Saiba como os agricultores no Egito Antigo faziam para bombear a água para irrigação.* (Irrigação.net) Acesso em 10 de 11 de 2022, disponível em https://www.irrigacao.net/outras_noticias/curiosidade-saiba-como-os-agricultores-no-egito-antigo-faziam-para-bombear-a-agua-para-irrigacao/

Testezlaf, R. (2017). *Irrigação: Métodos, Sistemas e Aplicações.* Campinas: FEAGRI.

Víqua. (s.d.). *Conheça as vantagens da irrigação automatizada | Blog ViQUA.* Acesso em 17 de 11 de 2022, disponível em Blog ViQUA: <https://www.viqua.com.br/blog/conheca-vantagens-da-irrigacao-automatizada>

Apêndice A

Código fonte utilizado na programação do Arduino MEGA 2560.

```
/*
# Universidade Federal de Viçosa - UFV
# Centro de Ciências Exatas e Tecnológicas - CCE
# Departamento de Engenharia Elétrica - DEL
# Protótipo de Sistema de Irrigação Automatizado
para Cafezais
# Desenvolvido por: Daniel Sossai Hupsel Celestino
# Matrícula: ES86299
*/

#include <Wire.h>
#include <RTCLib.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>
// Essa parte serve para definir as variáveis iniciais

RTC_DS3231 rtc;
LiquidCrystal_I2C lcd(0x3F,16,2);
//LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
int valvePins[] = {22, 26, 30, 34}; // pinos das válvulas
solenóide
int sensorPins[] = {A0, A1, A2, A3}; // pinos dos
sensores de umidade
int valveTime = 1 * 30; // tempo de ativação de cada
válvula em segundos
unsigned long valveStartTime[] = {0, 0, 0, 0}; // horário
de início da ativação de cada válvula
bool valveOn[] = {false, false, false, false}; // estado das
válvulas
int currentValve = -1; // válvula atualmente ativada
float RAIN = 0;

// Configuração do sensor DHT
#define RAIN A4
#define DHTPIN 47
#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

char daysOfTheWeek[7][16] = {"Domingo", "Segunda-
feira", "Terça-feira", "Quarta-feira", "Quinta-feira",
"Sexta-feira", "Sabado"};

String raindef;
// início do código principal

void setup() {
  Serial.begin(9600);
  Wire.begin();
  dht.begin();
  rtc.begin();
  lcd.init();
  lcd.backlight();

  lcd.setCursor(0, 0);
  lcd.print("Bem vindo");
  lcd.setCursor(0,1);
  lcd.print("Lendo sensores..");
  // lcd.print("Data: ");
  // lcd.print(rtc.now().day(), DEC);
  // lcd.print("/");
  // lcd.print(rtc.now().month(), DEC);
  // lcd.print("/");
  // lcd.print(rtc.now().year(), DEC);
  // lcd.setCursor(0, 1);
  // lcd.print("Hora: ");
  // lcd.print(rtc.now().hour(), DEC);
  // lcd.print(":");
  // lcd.print(rtc.now().minute(), DEC);
  // lcd.print(":");
  // lcd.print(rtc.now().second(), DEC);
  for (int i = 0; i < 4; i++) {
    pinMode(valvePins[i], OUTPUT);
    digitalWrite(valvePins[i], HIGH);
  }
  while(!Serial){
  }
  if(!rtc.begin()){
    Serial.println("Módulo RTC não encontrado");
    Serial.flush();
    abort();
  }
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

  if (rtc.lostPower()) {
    Serial.println("Módulo RTC ficou sem alimentação,
será realizado calibração.");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  }
  delay(5000);
}

void loop() {
  delay(500);

  // Medições dos sensores a cada 10 segundos
  static unsigned long previousMillis = 0;
  const unsigned long interval = 5000; // intervalo de
10 segundos entre as medições

  unsigned long currentMillis = millis();

  if (currentMillis - previousMillis >= interval) {
    previousMillis = currentMillis; // atualiza o valor de
previousMillis para o tempo atual

    DateTime now = rtc.now();
    float DHT_HUMIDITY = dht.readHumidity();
```

```

float DHT_Temperature = dht.readTemperature();
float Sensor_RAIN = analogRead(RAIN);
float moisture1 = analogRead(sensorPins[0]);
float moisture2 = analogRead(sensorPins[1]);
float moisture3 = analogRead(sensorPins[2]);
float moisture4 = analogRead(sensorPins[3]);

// Conversão de medições de sensores para
porcentagem
float moisture10 = (1 - moisture1/1023)*100;
float moisture20 = (1 - moisture2/1023)*100;
float moisture30 = (1 - moisture3/1023)*100;
float moisture40 = (1 - moisture4/1023)*100;
float Sensor_RAIN100 = (1 - Sensor_RAIN/1023)*100;
if (Sensor_RAIN100>50){
  raindef = "Forte";
}
else if(Sensor_RAIN100>20){
  raindef = "Fraca";
}
else{
  raindef = "Ausente";
}
Serial.println("  Sossai Irriga");
Serial.print("  Data: ");
Serial.print(rtc.now().day(), DEC);
Serial.print("/");
Serial.print(rtc.now().month(), DEC);
Serial.print("/");
Serial.println(rtc.now().year(), DEC);
Serial.print("  Hora: ");
Serial.print(rtc.now().hour(), DEC);
Serial.print(":");
Serial.print(rtc.now().minute(), DEC);
Serial.print(":");
Serial.println(rtc.now().second(), DEC);
Serial.print("    Dia da semana: ");

Serial.println(daysOfTheWeek[rtc.now().dayOfTheWeek()]);
Serial.print("  Temperatura: ");
Serial.print(DHT_Temperature);
Serial.println(" °C");
Serial.print("  Umidade do ar: ");
Serial.print(DHT_HUMIDITY);
Serial.println("%");
Serial.print("  Chuva: ");
Serial.println(raindef);
Serial.print("  Umidade do solo no setor 1: ");
Serial.println(moisture10);
Serial.print("  Umidade do solo no setor 2: ");
Serial.println(moisture20);
Serial.print("  Umidade do solo no setor 3: ");
Serial.println(moisture30);
Serial.print("  Umidade do solo no setor 4: ");
Serial.println(moisture40);
Serial.println(" ----- FIM -----");
Serial.println(" ");
if (currentValve == -1){
//  lcd.clear();
//  lcd.setCursor(3,0);
//  lcd.print("Irrigacao");
//  lcd.setCursor(3,1);
//  lcd.print("Concluida");
//  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Data: ");
  lcd.print(rtc.now().day(), DEC);
  lcd.print("/");
  lcd.print(rtc.now().month(), DEC);
  lcd.print("/");
  lcd.print(rtc.now().year(), DEC);
  lcd.setCursor(0,1);
  lcd.print("Hora: ");
  lcd.print(rtc.now().hour(), DEC);
  lcd.print(":");
  lcd.print(rtc.now().minute(), DEC);
  lcd.print(":");
  lcd.print(rtc.now().second(), DEC);
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Dia da semana: ");
  lcd.setCursor(0,1);

  lcd.print(daysOfTheWeek[rtc.now().dayOfTheWeek()]);
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Temperatura: ");
  lcd.setCursor(0,1);
  lcd.print(DHT_Temperature);
  lcd.print(" C");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Umidade do ar: ");
  lcd.setCursor(0,1);
  lcd.print(DHT_HUMIDITY);
  lcd.print("%");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Setor 1: ");
  lcd.print(moisture10);
  lcd.print("%");
  lcd.setCursor(0,1);
  lcd.print("Setor 2: ");
  lcd.println(moisture20);
  lcd.print("%");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Setor 3: ");
  lcd.println(moisture30);
  lcd.print("%");
  lcd.setCursor(0,1);
  lcd.print("Setor 4: ");
}

```

```

    lcd.println(moisture40);
    lcd.print("%");
    delay(3000);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Lendo Sensores...");
    delay(3000);
}
}
//-----//

for (int i = 0; i < 4; i++) {
    float moisture = analogRead(sensorPins[i]);
    float moisture100 = (1 - (moisture/1023))*100;
    float Sensor_RAIN1 = analogRead(RAIN);
    float Sensor_RAIN10 = (1 - Sensor_RAIN1/1023)*100;
    if (moisture100 < 40 && !valveOn[i] && Sensor_RAIN10 < 20) {
        if (currentValve == -1) {
            valveOn[i] = true;
            valveStartTime[i] = rtc.now().unixtime();
            digitalWrite(valvePins[i], LOW);
            currentValve = i;
            lcd.clear();
            lcd.setCursor(0,1);
            lcd.print("Umidade: ");
            lcd.print(moisture100);
            lcd.print("%");
            lcd.setCursor(0, 0);
            lcd.print("Setor ");

            lcd.print(i + 1);
            lcd.print(" ON ");
        }
    }

    if (valveOn[i]) {
        if (rtc.now().unixtime() - valveStartTime[i] >= valveTime) {
            digitalWrite(valvePins[i], HIGH);
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Setor ");
            lcd.print(i+1);
            lcd.print(" OFF");
            lcd.setCursor(0,1);
            lcd.print("Lendo sensores..");
            delay (3000);
            Serial.print(" Fim de ciclo de irrigação do setor ");
            Serial.println(i + 1);

            valveOn[i] = false;
            currentValve = -1;
        }
    }
}
}
}
}

```