

# Desenvolvimento de um Dispositivo *Stand-Alone* de Coleta de Dados de Interesse Agroecológico

Matheus Leal Cosmo Cardoso<sup>1</sup>, Sérgio Barros de Sousa<sup>1</sup>

<sup>1</sup>Universidade estadual do Piauí (UESPI) – Campus Professor Alexandre Alves de Oliveira  
Parnaíba – PI – Brazil

matheuscardoso.pro@gmail.com; sergiobarros@pq.cnpq.br

**Abstract.** *The irrigation activity is the second largest responsible for the consumption of fresh water in the world, behind only human consumption. As a result of recent technological advances, the Internet of Things is a technological paradigm that allows the interconnection of everyday objects with the internet. The application of the IoT paradigm can be observed in the most diverse sectors, such as in the development of technologies that assist in the realization of the data sensing of meteorological conditions for the most diverse purposes of study. This article discusses the development of a device for the collection and storage of agroecological data of a thermo-hygrometric nature that can be used in the most diverse scientific studies that can be performed from the possession of these data.*

**Resumo.** *A atividade de irrigação é a segunda maior responsável pelo consumo da água doce do mundo, atrás apenas do consumo humano. Como resultado dos recentes avanços tecnológicos, a Internet das Coisas é um paradigma tecnológico que permite a interconexão de objetos do dia a dia com a internet. A aplicação do paradigma de IoT pode ser observado nos mais diversos setores, como no desenvolvimento de tecnologias que auxiliem na realização do sensoriamento de dados de condições meteorológicas para os mais diversos fins de estudo. Este artigo aborda o desenvolvimento de um dispositivo para coleta e armazenamento de dados agroecológicos de natureza termo-higrométrica que poderá ser utilizado nos mais diversos estudos científicos que possa ser realizado a partir da posse destes dados.*

## 1. Introdução

Nas últimas décadas, discussões não só no contexto nacional, mas em escala global, vêm sendo levantadas a respeito do uso da água de forma eficiente em todas as suas áreas de utilização. Segundo dados constantes da Agência Nacional de Águas sobre a situação da água no mundo, apenas 2,5% do recurso é adequado ao consumo humano direto e à irrigação de plantações, dessa porcentagem citada, 69% (maior parte) é considerada de difícil acesso, 30% são águas subterrâneas e o restante (1%) encontram-se nos rios [ANA 2019b]. Por sua relevância para a vida, é muito importante a busca célere por soluções que sirvam de auxílio tornando o uso deste recurso eficiente e racional.

Uma das áreas de aplicação da água, a irrigação, é uma prática adotada pelos agricultores para suprir a deficiência parcial ou total da água em condições normais relacionadas a fatores climáticos. A cultura da irrigação no Brasil teve sua origem no final

do século XIX e início do século XX nas lavouras de arroz do Rio Grande do Sul e teve sua intensificação na década de 1970, devido à expansão da agricultura, as políticas de desenvolvimento regional e aos próprios benefícios advindos do desenvolvimento desta cultura observados na prática. Ainda segundo a Agência Nacional de Águas, o Brasil está ranqueado entre os 10 países com maior área equipada para irrigação do mundo, com dados estimados em 6,95 milhões de hectares equipados para a prática da cultura, números considerados pequenos pela agência frente ao potencial de expansão estimado de cerca de 76,2 milhões de hectares [ANA 2019a]. Para atingir o centro de todo esse potencial muitos avanços ainda são necessários para alcançar uma melhora na caracterização e monitoramento da atividade de irrigação, é necessário ainda que se faça um estudo aprofundado sobre o uso eficiente da água e a precisão da aplicação desta nas culturas.

Com a ascensão e disseminação das tecnologias de redes e automação, aliadas a consolidação de paradigmas como IoT (Internet of Things) e WSN (Wireless Sensors Network), tornou-se factível a possibilidade de combinação destes dois paradigmas no contexto de resolução de problemas por meio do monitoramento de ambientes tanto internos, quanto externos e também a possibilidade de controle de objetos de forma remota [Lima et al. 2018]. Dentro desta abordagem, vem tornando-se frequente o uso de dispositivos microcontroladores conectados a sensores como de temperatura e umidade, sensores de pressão, dentre outros, com a finalidade exclusiva de monitorar ambientes no qual, a necessidade de domínio dos dados captados por estes sensores, é um parâmetro importante e exerce influência direta na segurança e no modo de operação de equipamentos, seja em ambientes industriais ou agrícolas, proporcionando assim uma diretiva efetiva e econômica [Zhu et al. 2010].

A Internet das Coisas emergiu dos avanços de várias áreas como sistemas embarcados, microeletrônica, sensoriamento, entre outros. A IoT passou a receber bastante atenção, primeiramente, no âmbito acadêmico e passando, posteriormente, a ser difundida em larga escala no campo industrial, devido ao seu potencial de uso nas mais diversas áreas das atividades humanas. De uma forma bem resumida e introdutória, a Internet das Coisas é uma extensão da internet atual que possibilita objetos do dia a dia, como por exemplo, uma cafeteira, uma geladeira, etc, que possuam uma capacidade computacional mínima e também de comunicação, se conectarem a internet e poderem trocar informações entre si, com ou sem a interferência humana. Desta forma, conectados a internet, estes objetos podem ser acessados de forma remota ou ainda como provedores de serviços.

Portanto, como fora visto, diante dos dados e do contexto levantado, é factível que a agricultura do futuro exigirá da ciência de hoje e dos próximos anos a busca por soluções de baixo custo para o uso cada vez mais racional da água. O conhecimento científico gerado nas últimas décadas, torna possível o desenvolvimento de estudos e a implementação de tecnologias que possibilitem a utilização da água na agricultura de forma racional e eficiente, porém o que se enxerga é que até o momento existem poucas soluções que permitam mitigar este problema, e que existe ainda práticas que apresentam imprecisão e subjetividade na forma como são obtidos dados higrométricos e térmicos para as mais diversas finalidades, dentre elas, o objeto de motivação desta proposta, além de soluções que possuam um certa facilidade em serem disseminadas e que possuam baixo custo de implementação e manutenibilidade.

Sustentado nesta abordagem este artigo apresenta uma solução de baixo custo e baixo consumo energético que possa resolver o problema da obtenção de dados termohigrométricos, como temperatura e umidade do solo, tornando esse processo preciso e objetivo por meio da implementação de um dispositivo que realize a coleta automática e computadorizada destes dados integrando conceitos de IoT por meio da conexão entre microcontroladores e do uso de sensores e podendo assim ser utilizado para as mais diversas aplicações que tenham como objeto de estudo os dados coletados pelo dispositivo, dentre as quais vale destacar a mais importante delas que é a busca pela otimização do uso da água destinada à atividade de irrigação em lavouras.

## **2. Referencial Teórico**

### **2.1. Internet of Things (IoT)**

A Internet das Coisas ou Internet of Things (IoT), como conceito, passou e vem passando por constantes reformulações nas últimas décadas desde quando foi utilizado pela primeira vez em 1999 por Kevin Ashton, cofundador do Auto-ID Center do Massachusetts Institute of Technology (MIT), na época a visão de Internet of Things ainda era fundamentada na tecnologia de identificação por radiofrequência, do inglês Radio-Frequency Identification (RFID), desde então o paradigma vem amadurecendo expandindo consideravelmente o conjunto de tecnologias compreendidas e cenários de aplicação atendidos [Rozsa et al. 2017].

Segundo [Sri 2016], o paradigma de Internet of Things é um dos principais desenvolvimentos tecnológicos disruptivos dos últimos anos e pode ser considerado a próxima era no setor de tecnologia da informação, o que levaria o campo da tecnologia a novos patamares. Todo esse potencial permitiu o surgimento de uma gama de aplicações tais como coleta e monitoramento de dados de pacientes, sensoriamento de ambientes de difícil acesso, inóspitos, entre outras aplicações no qual se pode frisar ainda a coleta e monitoramento de dados agroecológicos para as mais diversas aplicações. Logo, pode-se afirmar que a IoT é um conjunto de diversas tecnologias que se complementam no sentido de suscitar a integração dos objetos no ambiente físico ao mundo virtual.

Em [Santos et al. 2016] são descritos os principais blocos básicos para a construção da IoT, partindo inicialmente do bloco de identificação do processo no qual é considerado por ele um dos mais importantes, pois é nele que por meio do emprego de tecnologias e protocolos de rede, serão identificados os objetos que irão funcionar com os mesmos. Em seguida tem-se o bloco de sensores/atuadores no qual se refere a forma como os sensores funcionam dentro do escopo dos dispositivos no qual eles são acoplados, destacando desde a tarefa dedicada a eles da coleta dos dados até o armazenamento dos mesmos.

Sequencialmente vem o bloco de comunicação que trata das diversas técnicas utilizadas para promover a conexão entre os objetos inteligentes, o autor ainda destaca algumas tecnologias empregadas no processo como, WiFi, Bluetooth, RFID, entre outras. Posterior ao bloco de comunicação tem-se o bloco de computação no qual é descrito a etapa que inclui a unidade de processamento que será responsável pela execução dos algoritmos locais nos dispositivos, um exemplo dessa unidade de processamento seria um microcontrolador, que será descrito em uma seção posterior. Por fim, tem-se os dois últimos blocos, a de serviço e o bloco da semântica que integrados aos blocos anteriores

mencionados formam a estrutura básica de blocos da construção de um dispositivo IoT [Rozsa et al. 2017].

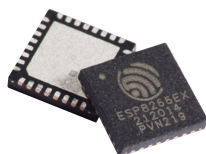
## 2.2. Microcontrolador

### 2.2.1. ESP8266 e NodeMCU

De acordo com [Silva 2006] um microcontrolador é um circuito integrado programável que contém todos os componentes básicos de um computador como CPU (Unidade Central de Processamento), memória para armazenamento de programas, memória de trabalho, portas de entrada e saída para comunicação com periféricos, sistema de controle de tempo interno e externo, conversor analógico digital, UART (Universal Asynchronous Receiver/Transmitter, Receptor/Transmissor Assíncrono Universal) de comunicação e outros. Cada microcontrolador possui uma aplicabilidade característica e específica, e fica a critério do projetista a escolha da melhor solução para seu projeto.

O ESP8266EX trata-se de um microcontrolador do tipo System-on-a-Chip (SoC) como pode ser observado na Figura 1, que possui Wi-Fi integrado que conta ainda com uma pilha TCP-IP completa fabricado pela empresa chinesa Espressif Systems [Espressif 2019].

O chip conta com um processador LX106 32-bit de arquitetura RISC (Reduced Instruction Set Computer, Computador com um Conjunto Reduzido de Instruções) com núcleo baseado no Tensilica Xtensa Diamond 106Micro, linha de processadores fabricados pela Tensilica, que opera sob uma frequência padrão de 80 MHz podendo ser configurada para operar em até 160 MHz, uma memória RAM de 128 KB no qual 32KB estão destinados ao conjunto de instruções do chip e o restante (96 KB) está destinado para dados, por fim, um armazenamento Flash QSPI externo de 512 KB a 4 MB podendo o mesmo ser configurado para uma capacidade de armazenamento de até 16 MB [Tensilica 2007].



**Figura 1. Wi-Fi Chip ESP8266EX**

Para facilitar o desenvolvimento de aplicações para o chip ESP8266EX foi criado o módulo NodeMCU. O NodeMCU funciona integrado com o módulo ESP-12 (Figura 2), responsável por realizar a intermediação entre o NodeMCU e o chip, a placa conta com 30 pinos de GPIO, possui um regulador de tensão de 3,3V AMS1117 com corrente máxima de 1A, interface Serial-USB, pinos para os protocolos de comunicação serial Inter-Integrated Circuit (I2C) e Serial Peripheral Interface (SPI), conversor analógico digital, etc [Advanced Monolithic Systems 2019]. O detalhamento das entradas e saídas GPIO da placa podem ser melhor observados na Figura 3.



Figura 2. Módulo Wi-Fi ESP-12

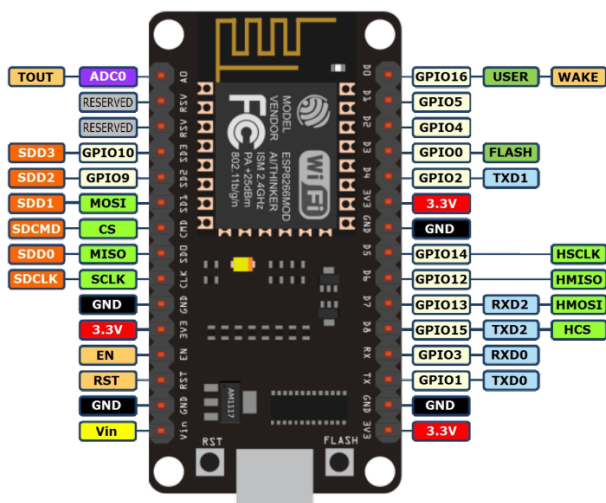


Figura 3. Placa ESP8266 NodeMCU

Em operação o ESP8266 pode ser configurado para funcionar em dois modos, são eles: Access Point e Client. No modo Access Point, ele funciona como um roteador, criando um ponto de acesso WiFi restrito por login e senha. Nesse modo o ESP8266 cria um servidor com IP (Internet Protocol, Protocolo de internet) aleatório ou pré-definido dependendo da programação realizada, esse servidor pode conter uma página web com informações dos componentes ligados ao ESP8266, por exemplo, dados captados por meio de leitura de sensores. No modo Client, o módulo estabelece uma conexão com uma rede WiFi escolhida, uma vez conectado também cria um servidor e todos os dispositivos conectados na mesma rede WiFi que o ESP8266 têm acesso a este servidor pelo endereço de IP. Esse servidor também pode conter uma página web e seu IP também pode ser aleatório ou pré-definido na programação. Ainda é possível realizar uma terceira configuração e fazer o ESP8266 trabalhar simultaneamente nos dois modos [Skraba et al. 2016].

### 2.3. Sensores

Segundo [Thomazini 2005], sensores são dispositivos sensíveis a formas de energias do ambiente em que estão inseridos, estas energias mensuradas podem ser luminosa, térmica, cinética, sempre relacionando uma grandeza física que precisa ser mensurada como, temperatura, pressão, velocidade, corrente, aceleração, posição, etc. São os componentes mais utilizados na automação em geral, seja no setor residencial ou industria. Os sensores podem ser de dois tipos, analógicos ou digitais, esta classificação é feita de acordo com a forma no qual o componente responde a variação da condição.

### 2.3.1. Sensores Analógicos

Os sensores do tipo analógico podem assumir qualquer valor em sua saída ao longo do tempo dentro de uma faixa de operação chamada de tensão. Qualquer grandeza física que assuma qualquer valor ao longo do tempo é caracterizada como uma grandeza analógica e são mensuradas por elementos sensíveis com circuitos eletrônicos não digitais [Thomazini 2005]. A Figura 4 ilustra a variação de uma grandeza analógica ao longo do tempo.

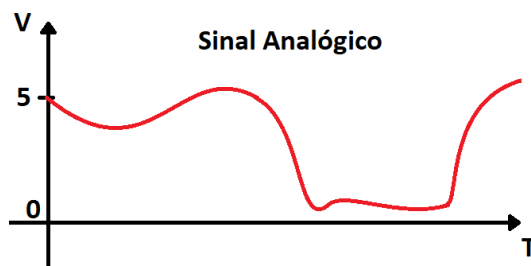


Figura 4. Gráfico sinal analógico

### 2.3.2. Sensores Digitais

Sensores digitais baseiam-se em níveis de tensão bem definidos, esse tipo de sensor pode assumir apenas dois valores em seu sinal de saída, esses valores podem ser descritos com alto (High) ou baixo (Low) que em binário é representado por 1(um) e 0(zero) respectivamente. As grandezas a serem medidas por esse tipo de sensor são convertidas no circuito de um transdutor para poderem assumir os valores discretizados (Figura 5) em sua saída [Patsko 2006].

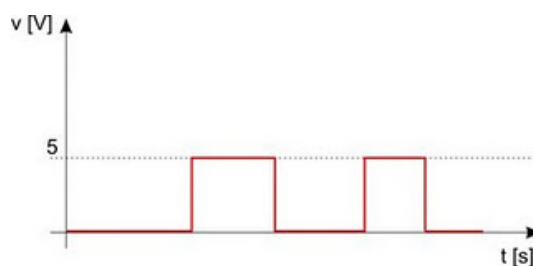


Figura 5. Gráfico sinal digital

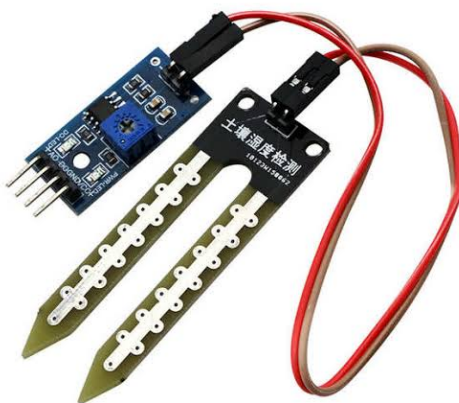
Neste artigo, para a implementação do dispositivo, foram utilizados dois sensores, o primeiro trata-se do sensor bmp280 (Figura 6) para realizar a captação de dados como temperatura, pressão e umidade atmosféricas. O segundo foi o módulo de sensoriamento FC-28 (Figura 7) para realizar a coleta de dados de umidade do solo.

O bmp 280 é um sensor digital de alta precisão, linearidade e estabilidade a longo prazo, baseado na tecnologia de sensoriamento Piezo-resistiva de pressão da Bosch. Este sensor conta com faixas de medição de pressão e temperatura consideráveis e o seu ponto forte reside no fato de o mesmo poder operar nos dois modos de interface digital, tanto no modo I2C quanto SPI [Sensortec 2018].



**Figura 6. Sensor bmp280 Bosch**

O Sensor Higrômetro FC-28 é do tipo analógico, podendo também funcionar de forma digital, utilizado para realizar a captação de umidade do solo, o sensor possui duas pads que atuam como um resistor variável que são inseridas no solo como uma espécie de sonda, dependendo da quantidade de água presente no solo, ou seja, quanto maior a umidade, maior será a condutividade registrada pelas pads do sensor, logo menor será a resistência, fazendo então com que, a partir do exposto anteriormente, o valor da umidade para aquele trecho de solo seja estabelecido [Emerson Cavagnino 2016].



**Figura 7. Sensor Higrômetro de Umidade do Solo FC-28**

### **3. Metodologia**

O trabalho em questão foi uma pesquisa de natureza empírica de caráter quantitativo no qual seguiu 2 (dois) eixos, priorizando a verificação da viabilidade da proposta e intencionando adequar os resultados esperados ao ambiente do agronegócio com baixo custo de implementação.

A metodologia empregada neste projeto foi realizada em 2 (duas) etapas, a primeira consistiu na codificação do firmware para o dispositivo em que foi realizada a implementação das instruções de leitura e armazenamento dos dados por parte dos sensores e microcontrolador. A segunda foi a de desenho e prototipação, no qual foi feito o desenho do modelo esquemático do circuito em que o firmware foi executado, na prototipação e construção do dispositivo a partir do modelo esquemático elaborado.

## 3.1. Codificação do Firmware

### 3.1.1. Leitura dos dados

A etapa metodológica inicial foi a codificação do firmware no qual foi realizada levando em consideração todos os componentes, microcontrolador e sensores, que foram utilizados para a prototipação do dispositivo, para isso foi empregado a utilização de bibliotecas para correta codificação e integração, por parte do firmware, de todos os componentes, principalmente por parte do sensor de temperatura e pressão bmp280 e do nodeMCU que requereram bibliotecas próprias para correto funcionamento.

O sensor de umidade do solo FC-28 não necessitou de nenhuma biblioteca em específico, bastando somente conectá-lo ao microcontrolador nas portas de alimentação (3V e GND) e na porta analógica (A0) (Figura 12). O microcontrolador para o qual o firmware foi desenvolvido foi o módulo NodeMCU (Figura 3), enquanto que os sensores utilizados foram dois, sensor de pressão e temperatura bmp280 e o sensor de umidade do solo FC-28.

O sensor de pressão e temperatura bmp280 realiza a leitura de dados de pressão atmosférica, temperatura e altitude, na codificação do firmware foi designado a este sensor apenas a leitura da temperatura. O sensor foi conectado ao microcontrolador utilizando a Interface Periférica Serial (SPI, da sigla em inglês), a interface I2C seria o ideal, porém a interface I2C do microcontrolador foi destinada ao Display LCD.

Na parte de codificação, o sensor bmp280 necessitou de duas bibliotecas específicas para seu funcionamento (trecho da linha 4 a 5 da Figura 8), as bibliotecas contém funções prontas e foram implementadas para dar facilidade durante o processo de codificação para atribuição das instruções a serem executadas pelo sensor. Em seguida foram definidas as portas no qual o sensor iria operar sob a interface SPI (linhas 7 a 10 da Figura 8)). Por fim foi realizado a criação de uma instância para uso das funções contidas nas bibliotecas no qual foram necessárias para a correta inicialização do sensor e para a realização da captação dos dados da temperatura (linhas 15 e 16 da figura 8)). A Figura 8 ilustra a codificação (bibliotecas, definições e funções) que foi necessária para configuração e correto funcionamento do sensor.

```
1 #include <Wire.h>
2 #include <SPI.h>
3 #include <ESP8266WiFi.h>
4 #include <Adafruit_Sensor.h> //Biblioteca do sensor bmp280
5 #include <Adafruit_BMP280.h> //Biblioteca do sensor bmp280
6
7 #define BMP_SCK D4 // SCK = SCL (D4)
8 #define BMP_SDI D5 // SDA = SDI (D5)
9 #define BMP_CS D6 // CS = CSB (D6)
10 #define BMP_SDO D7 // SHO = SHO (D7)
11
12 Adafruit_BMP280 bme; //Criação de um objeto chamado bme
13
14 void setup(){
15     //Inicializa o sensor bmp280
16     bme.begin();
17
18     //Realiza a leitura da temperatura
19     temp_value = bme.readTemperature();
20
21     //Grava o valor lido no ESP8266
22     writeTemperature(temp_value, PATH);
23 }
```

Figura 8. Código fonte sensor bmp280

Na codificação do sensor higrômetro FC-28, para realizar a leitura do nível de



umidade do solo foi criado uma função chamada **readMoisture** (linhas 3 a 12 da Figura 9), esta função faz a leitura do conversor analógico-digital do sensor e converte para um valor em porcentagem para ser gravado na memória do ESP8266.

Na linha 4 (Figura 9) foi criado uma variável para armazenar o valor lido no conversor analógico-digital, porta A0 (Figura 3), a partir da leitura deste valor foi feito o cálculo para determinar o valor percentual da umidade do solo. O valor do ADC (Analog-Digital Converter), no ESP8266, varia na faixa de 0 a 1023, sendo o valor 0 (zero) correspondente a 100% e o valor 1023 a 0% de umidade, dessa forma quanto maior o valor lido na entrada do ADC menor será a umidade registrada. Na linha 8 (Figura 9) está o código que realiza o cálculo para determinação do valor da umidade.

```
1 float readMoisture(){
2
3     float percentMoisture;
4     int ADC_Value = analogRead(A0); // 1023 em 3,3 volts
5
6     /* Transforma o valor lido na porta analogica
7      em um valor entre 0 e 100% */
8     percentMoisture = 100 * ((1023-(float)ADC_Value)/1023);
9
10    return percentMoisture;
11 }
12
13 umidade = readMoisture();
14
15 //Grava o valor lido na memoria do ESP8266
16 writeMoisture(umidade, PATH);
```

Figura 9. Código fonte para leitura da umidade do solo- Sensor FC-28

### 3.1.2. Gravação dos dados

Na parte de gravação dos dados foi utilizado o sistema de arquivos SPIFFS (SPI Flash File System), um sistema que foi desenhado para sistemas com memória SPI NOR flash, em dispositivos embarcados, ele é o sistema para manipulação de arquivos do ESP8266, o principal objetivo do sistema SPIFFS neste tipo de dispositivo é o fato de promover o menor uso possível de memória RAM, já que este recurso no NodeMCU, como já fora demonstrado, é muito limitado. O sistema possibilita a reserva de um espaço na memória flash para a criação e manipulação dos arquivos, essa alocação varia de 64kB até 3MB.

Para que fosse possível a criação e manipulação do arquivo na memória do NodeMCU, primeiro foi necessário realizar o upload do arquivo chamado **Dados\_Leitura.txt** para a memória flash do ESP8266 para somente então o arquivo poder ser manipulado via firmware. No código, para permitir a manipulação do arquivo foi necessário a importação da biblioteca **FS.h** do sistema SPIFFS, a biblioteca já é nativa do ambiente Arduino para o ESP8266, por meio dela é possível abrir o arquivo, realizar a leitura dos dados contidos no arquivo, gravar dados no arquivo, etc.

As funções **writeTemperature** (linha 22 da Figura 8) e **writeMoisture** (linha 16 da Figura 9), realizaram a gravação dos dados de temperatura e umidade do solo respectivamente. Junto aos dados coletados pelos sensores foi implementado também o armazenamento do horário em que a leitura foi feita e o tipo de leitura (Temperatura ou Umidade).

As duas funções são semelhantes em sua estrutura, para a implementação foi rea-

lizado a inclusão de dois parâmetros. O primeiro corresponde a um parâmetro correspondente ao valor do dado lido em cada sensor, o segundo, uma string informando o caminho do arquivo no qual os dados foram gravados. Para que os dados pudessem ser gravados no arquivo, foi codificada a instrução para abrir o sistema de arquivos por meio da função **openFS** (linha 2 da Figura 10 e 11). Com o sistema de arquivos aberto a instrução seguinte realizou a abertura do arquivo para adicionar os dados lidos, em seguida foi implementado uma estrutura de decisão (linhas 7 à 10 da figura 10 e 11) para saber se o arquivo foi aberto corretamente para ser realizada a gravação dos dados. Após a verificação bem sucedida o protocolo NTP foi atualizado para ser efetuado a gravação do horário da leitura de forma correta.

```

1 void writeTemperature(float dataTemperature, String path) {
2   openFS();
3   delay(1600);
4
5   File rFile = SPIFFS.open(path, "a");
6
7   if (!rFile) {
8     Serial.println("Erro_ao_abrir_arquivo!");
9   } else {
10    Serial.println("Salvando_dados_coletados...\n");
11
12    ntp.forceUpdate();
13
14    // Grava no arquivo as proximas 4 linhas de codigo
15    rFile.print("Temperatura."); // Leitura
16    rFile.print(dataTemperature); // Valor
17    rFile.print("_C.");
18    rFile.print(ntp.getFormattedTime()); // Horario
19  }
20  rFile.close();
21  SPIFFS.end();
22 }

```

**Figura 10. Função para gravar temperatura**

```

1 void writeMoisture(float dataMoisture, String path) {
2   openFS(); // Abre o sistema de arquivos
3   delay(1600);
4
5   File rFile = SPIFFS.open(path, "a");
6
7   if (!rFile) {
8     Serial.println("Erro_ao_abrir_arquivo!");
9   } else {
10    Serial.println("Salvando_dados_coletados...\n");
11
12    ntp.forceUpdate();
13
14    // Grava no arquivo as proximas 4 linhas de codigo
15    rFile.print("Umidade"); // Leitura
16    rFile.print(dataMoisture); // Valor
17    rFile.print("%");
18    rFile.print(ntp.getFormattedTime()); // Horario
19  }
20  rFile.close();
21  SPIFFS.end();
22 }

```

**Figura 11. Função para gravar Umidade**

Transcorrido todas as verificações, foram implementadas as instruções que realizaram o armazenamento das informações em uma nova linha do arquivo, por último foi codificado para que o arquivo e o sistema SPIFFS fossem encerrados (linhas 20 e 21 da Figura 10 e Figura 11), com a finalidade de evitar problemas na gravação dos dados.

Após ter finalizado a implementação das etapas de leitura e gravação dos dados a etapa final de implementação do firmware consistiu-se na codificação para promover a integração dos periféricos (LCD e teclado).

### 3.2. Esquema e Prototipação do Dispositivo

Após o término da codificação do firmware e para que fosse possível realizar os testes do correto funcionamento do mesmo foi dado início a etapa de desenho e prototipação do dispositivo.

Nesta etapa, os sensores e os componentes periféricos foram integrados ao microcontrolador. O desenho e a montagem foram realizados em uma protoboard, uma placa de ensaio que possui uma matriz de contatos que permite a construção de circuitos experimentais sem a necessidade de solda, permitindo com rapidez e segurança desde a alteração de posição de um determinado componente até sua substituição. Na Figura 12 é possível visualizar o modelo esquemático de como o protótipo foi montado na placa de ensaio com suas respectivas conexões enquanto que na Figura 13 está exposto o desenho do circuito esquemático do dispositivo. O planejamento e os desenhos do protótipo foram feitos com o auxílio da ferramenta Fritzing uma ferramenta de simulação dedicada para este propósito.

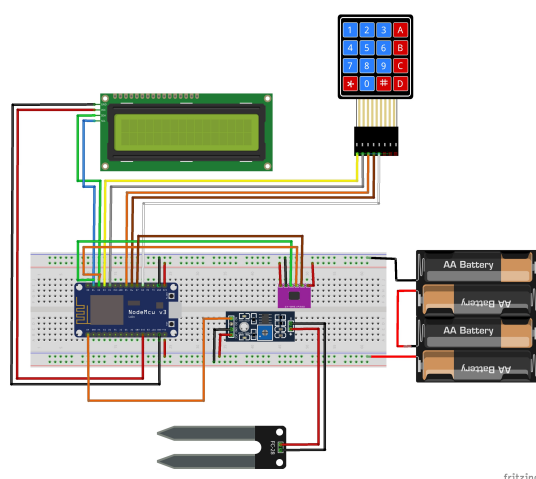
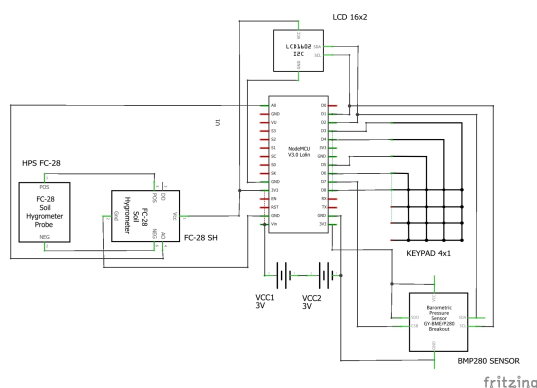


Figura 12. Modelo do protótipo

A integração dos componentes foi feita por partes, com o desenho do protótipo os componentes foram conectados de forma física. Inicialmente, foram integrados somente os sensores junto ao microcontrolador, então, foi realizado o upload do firmware de controle e efetuado o processo de coleta dos dados de forma automática sem auxílio de periféricos. Em seguida foi adicionado ao circuito o teclado matricial que permitiu o processo de coleta manual. Por último o display LCD foi integrado ao circuito. O protótipo foi alimentado por 4 (quatro) pilhas do tipo AA de 1,5V, perfazendo um total de 6V para a alimentação do circuito.

Com o número limitado de portas do ESP8266 foi integrado ao display LCD um módulo serial I2C para reduzir a quantidade de portas necessárias para conexão de um total de 16 conexões (sem o módulo) para um número máximo de 4 conexões após o integração do módulo. Com o display LCD operando com a interface I2C o sensor bmp280 foi configurado para operar com a interface do tipo SPI, para evitar qualquer tipo de curto na montagem do circuito. A Tabela 1 e 2 contém as configurações de conexão dos sensores bmp280 e FC-28 com as portas do microcontrolador.



**Figura 13. Circuito esquemático do Dispositivo**

**Tabela 1. Configuração das portas NodeMCU / bmp280**

NodeMCU	Sensor bmp280
3V	VCC
G	GND
GPIO2	SCL
GPIO14	SDA
GPIO12	CSB
GPIO13	SDO

**Tabela 2. Configuração das portas NodeMCU / FC-28**

NodeMCU	Sensor FC-28
3V	VCC
G	GND
A0	ADC0

Após a conclusão da codificação do firmware, da organização dos componentes e dos testes de bancada comprovarem a correta integração entre hardware e software, o dispositivo pôde ser então terminado.

## 4. Resultados e discussão

### 4.1. Testes e Validação

Na fase de testes e validação da proposta, foram realizados ensaios de leitura dos sensores integrados ao microcontrolador e aos demais componentes para verificação do correto funcionamento do dispositivo.

Os testes experimentais finais ocorreram após a construção do dispositivo no qual foi observado o correto funcionamento do mesmo da forma como fora planejado. o resultado da leitura e gravação dos dados pode ser analisado na Tabela 3 em que é exposto o resultado de 30 leituras realizadas com o dispositivo, os dados apresentados na tabela representam o índice da leitura, o tipo de leitura realizada, o valor lido por cada sensor e o horário que fora realizado a leitura respectivamente. Os dados são exibidos no servidor do ESP8266 acessado por meio do endereço IP gerado no servidor DHCP quando é realizada a conexão do NodeMCU a uma rede Wi-Fi.

**Tabela 3. Dados coletados dos sensores**

<b>DADOS LIDOS DOS SENSORES</b>			
<b>Nº</b>	<b>LEITURA</b>	<b>VALOR</b>	<b>HORÁRIO</b>
1	Umidade	46,73 %	15:46:51
2	Temperatura	32.21 °C	15:46:56
3	Umidade	25,90 %	16:09:36
4	Temperatura	31.96 °C	16:09:41
5	Umidade	29,03 %	16:10:35
6	Temperatura	31.99 °C	16:10:40
7	Umidade	35,97 %	16:14:10
8	Temperatura	32.85 °C	16:14:14
9	Umidade	37,24 %	16:24:28
10	Temperatura	32.22 °C	16:24:33
11	Umidade	34,67 %	16:40:31
12	Temperatura	29.99 °C	16:40:34
13	Umidade	33,92 %	16:44:30
14	Temperatura	30.11 °C	16:44:35
15	Umidade	33,72 %	16:45:15
16	Temperatura	30.15 °C	16:45:20
17	Umidade	33,72 %	16:46:32
18	Temperatura	30.10 °C	16:46:37
19	Umidade	33,53 %	16:53:53
20	Temperatura	29.87 °C	16:53:56
21	Umidade	31,57 %	11:06:46
22	Temperatura	30.65 °C	11:06:51
23	Umidade	32,36 %	11:08:21
24	Temperatura	31.16 °C	11:08:26
25	Umidade	35,00 %	11:10:16
26	Temperatura	31.72 °C	11:10:21
27	Umidade	41,54 %	11:27:45
28	Temperatura	33.37 °C	11:27:50
29	Umidade	46,34 %	11:46:50
30	Temperatura	32.86 °C	11:46:55

O valor dos dados de temperatura coletados pelo sensor bmp280 foram validados comparando o valor coletado pelo mesmo com os valores de temperatura disponíveis na plataforma climatempo no momento que o dado foi coletado pelo componente, os dados corresponderam em todas as comparações quando truncado, sem casas decimais, o valor registrado pelo sensor.

O sensor de umidade do solo FC-28 não necessitou de validação pois o valor da umidade auferida pelo sensor é tomado com base no cálculo do trecho de código da linha 8 na figura 9. para exemplificar como foi realizado o cálculo que determina a umidade do solo lida pelo sensor, tem-se um como suposição um valor lido na entrada ADC hipotético de 556, no cálculo realizado após o sinal de multiplicação obtêm-se um resultado, trun-

cado com 4 casas decimais, igual à 0.4565, após multiplicar este valor por 100 o resultado final calculado é de 45.65%, desta forma têm-se o valor da umidade correto para o trecho de solo determinado. O único problema associado com este sensor está no seu período de durabilidade, por não possuir um bom acabamento o sensor se torna suscetível a ação do fenômeno de oxidação que ocorre entre as hastes e faz com que o período de durabilidade do componente seja, em média, de apenas 3 à 4 meses, o que faz com que o correto funcionamento do sensor seja comprometido a partir do período mencionado.

Devido ao número limitado de portas do microcontrolador a forma de coleta dos dados teve de ser repensada pois o número restante de portas só permitiu que o teclado fosse integrado ao ESP8266 para realizar a coleta por meio de apenas uma tecla, a solução para o problema foi realizar o sensoriamento dos dados ao mesmo tempo com uma única instrução.

Após o término das fase de testes e validação o resultado final da pesquisa foi um dispositivo para a coleta e armazenamento de dados termo-higrométricos que pode ser observado na figura 14. O dispositivo foi integrado com conceitos de IoT e foi capaz de processar e armazenar as infamações de interesse agroecológico de forma correta e precisa.



**Figura 14. Dispositivo finalizado**

#### **4.2. Busca de Anterioridade**

O processo da busca teve como objetivo a verificação do estado da técnica por meio da varredura em bases de dados, nacionais e internacionais, tentando verificar um possível caráter inovador no dispositivo desenvolvido.

A busca de anterioridade apontou para um potencial inovador no dispositivo proposto. Inicialmente, a busca realizada na base de dados de patentes nacional do Instituto Nacional da Propriedade Industrial (INPI) foi realizada utilizando uma string de busca com palavras associadas ao dispositivo proposto, a string formada incluiu 04 (quatro) palavras **DISPOSITIVO, COLETOR, DADOS, TERMO-HIGROMÉTRICO**, a partir

de então foram utilizados filtros para a realização das buscas. Usando o filtro no qual todas as palavras estivessem contidas no título ou no resumo das patentes, nenhum resultado foi encontrado. Utilizando o filtro que buscava a expressão exata no resumo ou no título, utilizando a string informada, o resultado foi o mesmo. Quando o filtro foi alterado para buscar qualquer uma das palavras no resumo ou no título, a pesquisa retornou um total de 81.205 patentes. Inicialmente o que se observou foi que destas 81.205 grande parte dos resultados estavam associados com as palavras **DISPOSITIVO** e **COLETOR**, então decidiu-se reduzir a string de busca a somente uma palavra, **TERMOMIGROMÉTRICO**, e mantendo o filtro de busca a pesquisa retornou 2(dois) resultados, porém, o equipamento encontrado não era semelhante e nem estava associado com o propósito e finalidade do dispositivo proposto.

Com o resultado obtido na busca na base nacional de patentes, o passo seguinte foi realizar pesquisas nas bases internacionais. A primeira base a ser realizado buscas foi a base de patentes Norte-Americana USPTO, o sistema de buscas nesta base é diferente do nacional pois não utiliza filtro, sua forma de busca se deu por meio de expressões lógicas contendo as palavras de interesse associadas ao dispositivo, a expressão final para a realização da busca foi a seguinte (**((Device OR equipment) AND Data) AND (Termohygrometric OR Agroecological)**), a pesquisa trouxe 02(dois) resultados, mas os dois tratavam-se de métodos de fertilização, não se assemelhando em nada com o objeto desta proposta. Por fim, a última base de dados a ser realizado as buscas foi a base Europeia da Spacenet, o sistema de busca nesta base é semelhante ao da USPTO, logo a expressão utilizada foi a mesma, a pesquisa retornou 14(quatorze) patentes encontradas. Do total encontrado 10(dez) estavam associadas a métodos e técnicas agrícolas de fertilização e criação de adubos e o restante consistiam em dispositivos ou equipamentos para realizar a seleção de plantas. A partir dos resultados da busca pode-se afirmar que existe um potencial de inovação no objeto da proposta.

## 5. Conclusão

Neste trabalho foram abordados questões como o uso ineficiente da água destinada à prática da atividade de irrigação nas lavouras e como a aplicação ineficaz do recurso impacta diretamente à pouca reserva disponível para o consumo humano, por exemplo, assim como a busca por soluções que permitam tornar a prática desta atividade objetiva e eficiente, sustentado na ciência até aqui desenvolvida e com o auxílio de ferramentas tecnológicas que permitiram o desenvolvimento da solução aqui apresentada.

Foi possível concluir por meio do desenvolvimento deste dispositivo, com princípios baseados em IoT, que aplicações microcontroladas para realização de sensoriamento de ambientes e coleta automatizada de dados são viáveis com a utilização de poucos componentes possibilitando o desenvolvimento de soluções de baixo custo, necessitando apenas dos componentes necessários para a construção do dispositivo em si, das ferramentas e bibliotecas corretas para a implementação do firmware de controle e ainda o auxílio de plataformas que facilitam no desenvolvimento possibilitando a implementação rápida e eficaz de aplicações para este tipo de dispositivo.

A solução cumpriu com os objetivos propostos tendo ao final do projeto o esboço de um dispositivo capaz de coletar dados de forma automatizada, com objetividade e precisão, para o uso diário no auxílio as mais diversas aplicações que tenham como objeto

de análise dados termo-higrométricos.

## Referências

- Advanced Monolithic Systems, I. (2019). *AMS 1117, 1A LOW DROPOUT VOLTAGE REGULATOR*. Disponível em <http://www.advanced-monolithic.com/pdf/ds1117.pdf>.
- ANA (2019a). *Atlas Irrigação: Uso da água na agricultura irrigada*. Agência Nacional de Águas. Disponível em <http://atlasirrigacao.ana.gov.br/>.
- ANA (2019b). *Situação da água no mundo*. Disponível em <https://www.ana.gov.br/panorama-das-aguas/agua-no-mundo>.
- Bajrami, X. and Murturi, I. (2018). An efficient approach to monitoring environmental conditions using a wireless sensor network and NodeMCU. *e & i Elektrotechnik und Informationstechnik*, 135(3):294–301. <https://doi.org/10.1007/s00502-018-0612-9>.
- Emerson Cavagnino, F. C. (2016). Irrigador eco-eficiente para plantações. *Revista Ciência e Tecnologia*, 19(35). <http://www.revista.unisal.br/sj/index.php/123/article/view/539>.
- Espressif (2019). *ESP8266 Technical Reference*. Disponível em [https://www.espressif.com/sites/default/files/documentation/esp8266-technical\\_reference\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf).
- Fachini, M. P., Mesquita, N. P., Oliveira, R. P., and de França, P. G. (2017). Internet das coisas: Uma breve revisão bibliográfica. *Conexões - Ciência e Tecnologia*, 11(6):85. <https://doi.org/10.21439/conexoes.v11i6.1007>.
- Kamienski, C., Soininen, J.-P., Taumberger, M., Dantas, R., Toscano, A., Cinotti, T. S., Maia, R. F., and Neto, A. T. (2019). Smart water management platform: IoT-based precision irrigation for agriculture. *Sensors*, 19(2):276. <https://doi.org/10.3390/s19020276>.
- Lima, M., Alves, F., and Juca, S. (2018). Sistema iot para monitoramento de temperatura e umidade ambientes e acionamento remoto de cargas. In *Anais da IV Escola Regional de Informática do Piauí*, pages 232–237, Porto Alegre, RS, Brasil. SBC. <https://sol.sbc.org.br/index.php/eripi/article/view/5173>.
- Patsko, L. F. (2006). Tutorial—aplicações, funcionamento e utilização de sensores. *Maxwell Bohr: Instrumentação eletrônica*, page 84.
- Rozsa, V., Dutra, M. L., Pinto, A. L., and Muriel-Torrado, E. (2017). O paradigma tecnológico da internet das coisas e sua relação com a ciência da informação. *Informação amp; Sociedade: Estudos*, 27(3). <https://periodicos.ufpb.br/ojs2/index.php/ies/article/view/36975>.
- Santos, B. P., Silva, L., Celes, C., Borges, J. B., Neto, B. S. P., Vieira, M. A. M., Vieira, L. F. M., Goussevskaia, O. N., and Loureiro, A. (2016). Internet das coisas: da teoria à prática. *Minicursos SBRC-Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.



- Sensortec, B. (2018). *BMP 280, Digital Pressure Sensor*. Disponível em <https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bmp280-ds001.pdf>.
- Silva, R. A. (2006). *Programando Microcontroladores PIC*. [https://books.google.com.br/books/about/Programando\\_Microcontroladores\\_Pic\\_Lingu.html?id=48mZZwEACAAJ&redir\\_esc=y](https://books.google.com.br/books/about/Programando_Microcontroladores_Pic_Lingu.html?id=48mZZwEACAAJ&redir_esc=y).
- Skraba, A., Kolozvari, A., Kofjac, D., Stojanovic, R., Stanovov, V., and Semenkin, E. (2016). Streaming pulse data to the cloud with bluetooth LE or NODEMCU ESP8266. In *2016 5th Mediterranean Conference on Embedded Computing (MECO)*. IEEE. <https://doi.org/10.1109/meco.2016.7525798>.
- Sri, T. (2016). A review on the state of art of internet of things. *INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER AND COMMUNICATION ENGINEERING*, 5:38.
- Sundmaeker, H., Guillemin, P., Friess, P., and Woelfflé, S. (2010). Vision and challenges for realizing the internet of things. *Cluster of European Research Projects on the Internet of Things, European Commission*.
- Tensilica, I. (2007). *Tensilica Unveils Diamond Standard 106Micro Processor: Smallest Licensable 32-bit Core*. Disponível em <https://ip.cadence.com/news/243/330/Tensilica-Unveils-Diamond-Standard-106Micro-Processor-Smallest-Licens>
- Thomazini, D. (2005). *Sensores industriais: fundamentos e aplicações*. Saraiva Educação SA.
- TongKe, F. (2013). Smart agriculture based on cloud computing and IOT. *Journal of Convergence Information Technology*, 8(2):210–216. <https://doi.org/10.4156/jcit.vol18.issue2.26>.
- Zhu, Q., Dongmei, Z., and Xunwen, S. (2010). Distributed remote temperature monitoring and acquisition system based on CAN bus. In *2010 Prognostics and System Health Management Conference*. IEEE. <https://doi.org/10.1109/phm.2010.5413439>.