

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO DE ESPECIALIZAÇÃO EM INTERNET DAS COISAS

RUDY ERNESTO CAMARGO PEÑARANDA

**SISTEMA DE IRRIGAÇÃO AUTOMATIZADO PELA UMIDADE DO
SOLO**

MONOGRAFIA DE ESPECIALIZAÇÃO

CURITIBA

2018

RUDY ERNESTO CAMARGO PEÑARANDA

SISTEMA DE IRRIGAÇÃO AUTOMATIZADO PELA UMIDADE DO SOLO

Monografia de Especialização, apresentada ao Curso de Especialização em Internet das Coisas, do Departamento Acadêmico de Eletrônica – DAELN, da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Especialista.

Orientador: Prof. Dr. Ohara Kerusauskas Rayel

CURITIBA

2018



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Curitiba

Diretoria de Pesquisa e Pós-Graduação
Departamento Acadêmico de Eletrônica
Curso de Especialização em Internet das Coisas



TERMO DE APROVAÇÃO

SISTEMA DE IRRIGAÇÃO AUTOMATIZADO PELA UMIDADE DO SOLO

por

RUDY ERNESTO CAMARGO PEÑARANDA

Esta monografia foi apresentada em 27 de Novembro de 2018 como requisito parcial para a obtenção do título de Especialista em Internet das Coisas. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Dr. Ohara Kerusauskas Rayel
Orientador

Prof. M.Sc. Danillo Leal Belmonte
Membro titular

Prof. M.Sc. Omero Francisco Bertol
Membro titular

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso -

RESUMO

PEÑARANDA, Rudy Ernesto Camargo. **Sistema de irrigação automatizado pela umidade do solo**. 2018. 39 p. Monografia de Especialização em Internet das Coisas, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2018.

Este projeto tem como finalidade mostrar como se pode desenvolver uma aplicação com as novas tecnologias do mundo de Internet das Coisas, programar um microcontrolador hoje é mais comum, porém desenvolver um sistema de redes sem fio é ainda novo. Contiki é ainda um software pouco conhecido embora tem muitas utilidades que podem fornecer grande quantidade de ajuda para nosso projeto. Na área dos cultivos é necessário criar novas tecnologias para ajudar aos trabalhadores da área. O protocolo MQTT e MQTT-SN estão sendo utilizados por grandes empresas para criar novas aplicações para acrescentar a produtividade sobre uma rede de sensores, por isso tem muitos sistemas com este tipo de protocolos. A placa CC2650 tem sido muito utilizada e é uma boa opção para quem quer desenvolver um projeto de Internet das Coisas.

Palavras-chave: Internet das Coisas. Contiki. MQTT. MQTT-SN. CC2650.

ABSTRACT

PEÑARANDA, Rudy Ernesto Camargo. **Automated irrigation system for soil moisture**. 2018. 39 p. Monografia de Especialização em Internet das Coisas, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2018.

This Project have the point of showing how to develop an application with new technologies of internet of things, programming a controller today is more common but develop an unwired system is new. Contiki is not wellknown software but it has a lot utilities that can give a lot of help to our project. In agriculture is necessary to create new technologies that could help the people who work in that area. The MQTT and MQTT-SN protocols are being used in bigger companies that create new applications that are increasing the productivity in sensor networks. because of that they have a lot of systems working with those protocols. The CC2650 board have a los of utilities and it is a good option for who want to develop a project in internet of thing.

Keywords: Internet of Things. Contiki. MQTT. MQTT-SN. CC2650.

LISTA DE FIGURAS

Figura 1 - Irrigação da banana	9
Figura 2 - Endereço IPv6	13
Figura 3 - MQTT.ORG.....	15
Figura 4 - Configurações do MQTT e MQTT-SN.....	16
Figura 5 - Logo do Contiki	17
Figura 6 - Controlador CC2650	18
Figura 7 - Portas do controlador CC2650.....	19
Figura 8 - Logo do Code Composer	20
Figura 9 - Diagrama do sistema	21
Figura 10 - Sensor de umidade do solo	22
Figura 11 - Shield sensor de umidade do solo	22
Figura 12 - Irrigador de água.....	23
Figura 13 - Shield válvula solenoide água 12V/ 3/4	24
Figura 14 - Rede mesh em Zigbee.....	26
Figura 15 - Área do cultivo	26
Figura 16 - Exemplo de um tópico.....	27
Figura 17 - Troca de mensagens	28
Figura 18 - Sistema de comunicação	29
Figura 19 - Mensagem MQTT-SN.....	30
Figura 20 - Mensagem PUBLISH.....	30
Figura 21 - Mensagem de umidade.....	31
Figura 22 - Mensagem SUBSCRIBE.....	32
Figura 23 - Monitor serial	34
Figura 24 - Irrigação parada.....	35
Figura 25 - Pouca irrigação	36
Figura 26 - Irrigação completa.....	37

LISTA DE TABELAS

Tabela 1 - Funcionamento dos sensores	33
---	----

LISTA DE SIGLAS

6LoWPAN	<i>IPv6 over Low power Wireless Personal Area Networks</i>
IEEE	<i>The Institute of Electrical and Electronics Engineers</i>
IP	<i>Internet Protocol</i>
IPsec	<i>Internet Protocol security</i>
IPv4	<i>Internet Protocol version 4</i>
IPv6	<i>Internet Protocol version 6</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
MQTT-SN	<i>MQTT for Sensor Network</i>
pub/sub	<i>publish/subscribe</i>
QoS	<i>Quality of Service</i>
TCP	<i>Transport Control Protocol</i>
TCP/IP	<i>Transmission Control Protocol / Internet Protocol</i>
UDP	<i>User Datagram Protocol</i>

SUMÁRIO

1 INTRODUÇÃO	8
1.1 CONTEXTUALIZAÇÃO.....	8
1.2 PROBLEMA.....	8
1.3 OBJETIVOS.....	10
1.3.1 Objetivo Geral.....	10
1.3.2 Objetivos Específicos.....	10
1.4 JUSTIFICATIVA.....	10
1.5 ESTRUTURA DO TRABALHO	11
2 FUNDAMENTAÇÃO TEÓRICA	12
2.1 INTERNET DAS COISAS (INTERNET OF THINGS).....	12
2.2 INTERNET PROTOCOL VERSION 6 (IPV6).....	13
2.3 MQTT-SN.....	14
2.4 CONTIKI	16
3 DESCRIÇÃO DAS PARTES DO SISTEMA	18
3.1 DISPOSITIVO DE CONTROLE	18
3.2 SOFTWARE.....	19
3.3 CONFIGURAÇÃO DA REDE.....	20
3.3.1 Broker MQTT-SN	20
3.4 SENSORES	21
3.4.1 Sensor de Umidade do Solo	21
3.4.2 Atuador de Irrigação do Solo	23
4 DESENVOLVIMENTO.....	25
4.1 ZIGBEE.....	25
4.2 IEEE 802.15.4.....	26
4.3 ORGANIZAÇÃO DOS TÓPICOS PUBLICADOS.....	26
4.3.1 Formato das Mensagens	30
4.4 MENSAGEM PUBLISH.....	30
4.5 MENSAGEM SUBSCRIBE	32
4.6 INTERPRETAÇÃO DOS RESULTADOS.....	33
4.6.1 Testes	33
4.6.1.1 Irrigação parada.....	34
4.6.1.2 Pouca irrigação	35
4.6.1.3 Irrigação completa.....	36
5 CONCLUSÕES	38
REFERÊNCIAS.....	39

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Os protocolos de sensores de redes sem fio estão sendo desenvolvidos com mais frequência, porque o mundo atual já está acostumado aos dispositivos sem fio como os aparelhos celulares de última geração, porém muitos serviços ainda não estão acostumados a estas mudanças que podem fornecer a todo tipo de negócios ajuda para desenvolver uma melhor produtividade, além disso com ajuda do software livre que diminui os gastos.

A Internet das Coisas tem como objetivo ajudar a conectar as pessoas com objetos que dão informação sobre algum acontecimento, que pode ser uma queda de energia, umidade ou excesso de chuva podem ser transmitidos para o usuário sem que ele esteja na locação, tudo isso por meio de sensores.

O protocolo de mensagens *Message Queuing Telemetry Transport* (MQTT) é projetado baseado em *Transmission Control Protocol/Internet Protocol* (TCP/IP), o qual usa baixo consumo de banda de rede e com poucos requisitos de hardware, além disso tem muitas aplicações dentro das redes baixa banda de dados.

O protocolo tem duas partes: a) *cliente* e b) *broker*, funcionam com a pilha TCP/IP para o seu funcionamento. O MQTT utiliza o paradigma *publish/subscribe* (pub/sub) para a troca de mensagens. O *broker* trabalha no meio como responsável por receber e enviar as mensagens recebidas dos *publishers* para os *subscribers*. O *publisher* se conecta ao *broker* e publica as mensagens, ao mesmo tempo o *subscriber* também se conecta ao *broker* para receber as mensagens.

1.2 PROBLEMA

Segundo o lido em um artigo, a região Noroeste do estado de São Paulo é importante área de cultivo da fruta para a cidade de São Paulo, por isso é necessário ter um bom plano de trabalho para ajudar a produção de produtos agrícolas que servem para a alimentação de milhões de pessoas.

Um exemplo é a produção da banana nanica, onde as plantas são desbrotadas e a plantação tem que ser adubada com frequência recebendo muita água. Tem um investimento de R\$ 70 mil só em aspersores, de acordo com os estudos sobre esta fruta, se sabe que usa mais água que em outras plantações. A irrigação (Figura 1) e feita durante a noite, quando a energia é mais barata (GLOBO, 2018).

Figura 1 - Irrigação da banana



Fonte: Globo (2018).

A área atualmente tem muitos problemas com as chuvas, por isso usam agora água dos açudes e poços artesianos que garante a colheita durante o ano inteiro. O projeto tem por objetivo permitir que diferentes tipos de cultivos agrícolas sejam monitoradas e, com os parâmetros adequados para determinada área, esta seja irrigada de maneira autônoma porque segundo as pesquisas a ausência de chuvas não é o único fato causador do déficit hídrico no campo. O estresse hídrico é uma condição quando uma planta busca ajustar entre a absorção de água e a transpiração. Toda a dinâmica da água no sistema solo-planta-atmosfera ocorre em função da demanda evaporativa do local. A criação deste ambiente propício para os cultivos conseqüentemente proporcionará um aumento na produção dos alimentos gerados por elas.

1.3 OBJETIVOS

Diante das necessidades que se tem na área do agronegócio, se tem que ajudar a ter mais controle da água que se usa porque a escassez da água e a estiagem em certas regiões do país, pode afetar a toda a população.

1.3.1 Objetivo Geral

Desenvolver um sistema automatizado de irrigação que possa ajudar aos fazendeiros fornecendo água a seus cultivos sem desperdícios.

1.3.2 Objetivos Específicos

- Permitir que diferentes tipos de cultivos agrícolas tenham irrigação monitorada e autônoma com os parâmetros adequados para determinado cultivo, garantindo assim um ambiente onde a água não vai ser desperdiçada, assim a criação deste ambiente é propício para os cultivos conseqüentemente proporcionará um aumento na produção dos alimentos gerados;
- Adaptar um modelo específico que atenda aos requisitos do protocolo *MQTT for Sensor Network* (MQTT-SN) para redes de sensores sem fio;
- Usar todas as ferramentas necessárias de desenvolvimento para fazer um desenho acessível para que qualquer fazendeiro possa entender;
- Fazer um desenho que se possa diminuir os custos que atualmente se tem no mercado de aspersores industriais.

1.4 JUSTIFICATIVA

Atualmente no mundo 60% do consumo de água é realizado pela irrigação de cultivos agrícolas. No Brasil, o uso da água pela irrigação chega a 75% do total consumido. Mundialmente, apenas 37% da água aplicada no solo pela irrigação é

realmente aproveitada pelos cultivos. Segundo esses dados se tem que criar novas estratégias para fornecer água com os cuidados já previstos.

Algumas outras características que levaram a desenvolver este projeto é porque MQTT-SN se otimiza para implementações utilizando dispositivos de baixo custo, operados por bateria com recursos limitados de processamento e armazenamento e a baixa taxa de transmissão.

1.5 ESTRUTURA DO TRABALHO

Esta monografia de especialização está dividida em 5 (cinco) seções. Nesta primeira seção foi introduzido o assunto tema do trabalho e também foram abordados a motivação e os objetivos geral e específicos da pesquisa, a justificativa e a estrutura geral do trabalho.

Já na segunda seção: Fundamentação Teórica, será abordado toda a teoria que é a base do projeto.

A seguir na terceira seção: Descrição das Partes do Sistema, será abordado toda a informação sobre as ferramentas que serão utilizadas no desenvolvimento/implementação do projeto proposto.

Na quarta seção: Desenvolvimento, nesta parte tem-se o funcionamento do projeto.

Por último na quinta seção: Conclusões, onde se faz uma análise dos resultados do projeto.

2 FUNDAMENTAÇÃO TEÓRICA

Atualmente tem-se implementações de redes que podem coletar informações e/ou controlar os objetos do mundo físico através de vários dispositivos instalados como sensores. Estes dispositivos tem capacidade de dar solução e suportar interação entre coisas ou entre humanos e coisas, tudo por meio de uma rede sem fio.

2.1 INTERNET DAS COISAS (INTERNET OF THINGS)

O nome vem da expectativa de interconectar milhões de ilhas de redes inteligentes, permitindo o acesso à informação não apenas a qualquer momento e em qualquer lugar, mas também usando “qualquer coisa” e “qualquer pessoa”.

Os dispositivos de detecção, identificação e posicionamento estão dotados de um endereço *Internet Protocol* (IP) e se tornam objetos inteligentes, capazes de se comunicar não apenas com outros objetos inteligentes, mas também com seres humanos com a expectativa de alcançar áreas que nunca poderiam ser alcançadas sem os avanços feitos nas tecnologias de sensoriamento, identificação e posicionamento.

Embora possam ser descobertos e consultados globalmente, esses objetos inteligentes também podem descobrir e interagir com entidades externas consultando humanos, computadores e outros objetos inteligentes. Os objetos inteligentes também podem obter inteligência ao fazer ou ativar decisões relacionadas ao contexto, aproveitando os canais de comunicação disponíveis para fornecer informações sobre si mesmos e, ao mesmo tempo, acessar informações que foram agregadas por outros objetos inteligentes.

Estes são os três fundamentos de interesse:

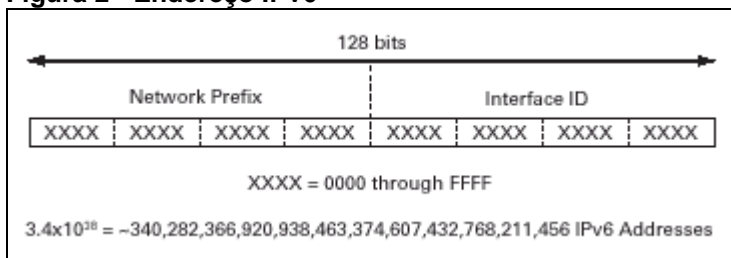
- a. Rede sem fio: A conectividade sem fio permite recuperar dados em tempo real a partir de locais de difícil acesso. Isso também torna o sistema de monitoramento menos intrusivo em locais onde os fios perturbariam a operação normal do ambiente a ser monitorado.

- b. Reduz os custos de instalação: estima-se que a tecnologia sem fio poderia eliminar até 80% desse custo;
- c. Baixa potência: Para funcionar por vários meses, os motes devem usar rádios e processadores de baixa potência e implementar esquemas eficientes de energia. O processador deve entrar no modo de espera o maior tempo possível e a camada de acesso médio deve ser projetada de acordo.

2.2 INTERNET PROTOCOL VERSION 6 (IPV6)

O endereçamento *Internet Protocol version 6* (IPv6) tem a seguinte forma, como pode-se observar na Figura 2.

Figura 2 - Endereço IPv6



Fonte: Maio, Kling e Almdeira (2018).

O uso de 128 bits para endereços traz alguns benefícios, são eles:

- Fornecer muito mais endereços, para satisfazer as necessidades atuais e futuras, com amplo espaço para inovação;
- Simplificar os mecanismos de configuração automática de endereços;
- Gerenciamento e delegação de endereços mais fácil;
- Espaço para mais níveis de hierarquia e para agregação de rotas;
- Capacidade de fazer *end to end* com *Internet Protocol security* (IPsec).

Os endereços IPv6 são classificados nas seguintes categorias, que também existem no *Internet Protocol version 4* (IPv4):

- *Unicast*: ou um para um;
- *Multicast*: ou um para muitos;
- *Anycast*: ou um para mais perto;

- Reservado.

A categoria *Unicast* é utilizada para enviar um pacote da origem para um único destino, eles são os mais comuns. Já a categoria *Multicast* é utilizada para enviar um pacote da origem para vários destinos. Isso é possível por meio de roteamento *Multicast* que permite que os pacotes sejam replicados em alguns locais. A categoria *Anycast* é utilizada para enviar um pacote da origem para o destino mais próximo de um conjunto deles. Por último, a categoria do estado de reservado serve para endereços ou grupos especiais, por exemplo, endereços a serem usados em documentação e exemplos.

2.3 MQTT-SN

Nesse caso usou-se o *MQTT for Sensor Network* (MQTT-SN), que é uma opção avançada para Internet das Coisas (ou *Internet of Things* - IoT), especial para redes sem fio. MQTT-SN trabalha com *User Datagram Protocol* (UDP) e, que trabalha exclusivamente com redes de sensores sem fio. O broker que tem a função de fornecer as mensagens entre os clientes ou terminais onde estão os sensores sem fio.

A identificação das mensagens é feita através de tópicos (*topics*) como MQTT, o modelo trabalha igual a MQTT com (*publisher/subscriber*), onde os clientes podem se inscrever (*subscribe*) em múltiplos tópicos e recebem todas as mensagens enviadas a este tópico, enquanto eles já estão inscritos podem enviar (*publish*) diversos tópicos para o *broker*. Existe uma organização de software livre que trabalha com MQTT e MQTT-SN como pode ser observado na Figura 3.

Figura 3 - MQTT.ORG



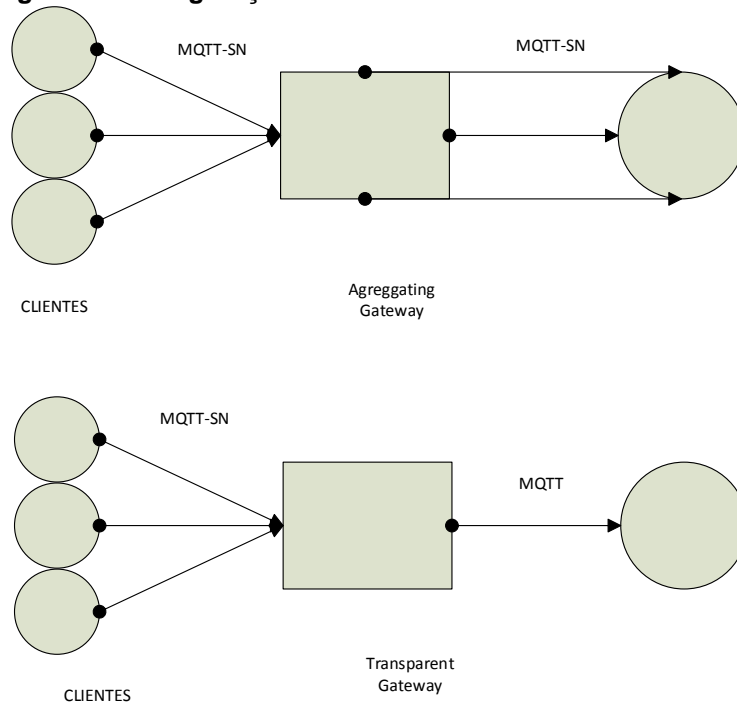
- On top of TCP/IP
- Publish/Subscribe messaging pattern
- Message Broker distributes topics to clients
- Topics are UTF-8 string-based with hierarchical structure
- No direct connection between clients
- Quality of Service
- Retain-Flag: new subscribed clients will received last value
- Last Will: notify other clients when disconnected ungracefully
- KeepAlive: ping request messages to the bróker
- Clients have to know beforehand the structure of the data published to a topic
- MQTT is data-agnostic

Fonte: Colina et al. (2016, p. 198).

Algumas caraterísticas do modelo MQTT-SN, são as seguintes:

- *Quality of Service* (QoS) de nível 1;
- Mensagem de *Will and Testament*, para desconexão inesperada;
- Mensagem de *Persistence/Retained* é armazenada no *broker* e enviada a todo nó que se registrar no tópico;
- Um procedimento de descoberta de servidores/*gateways* foi adicionado para informar aos clientes sobre o endereço de tais dispositivos e evitar múltiplas conexões. Na Figura 4, podem ser observadas as diferenças dos servidores/*gateways*, no MQTT têm uma porta de saída, já no MQTT-SN têm-se várias.

Figura 4 - Configurações do MQTT e MQTT-SN



Fonte: Stanford-Clark e Truong (2013, p. 6).

O MQTT-SN foi originalmente desenvolvido para rodar com Zigbee que é um consórcio industrial aberto com o objetivo de definir um padrão de comunicação aberto e global para redes de sensores sem fio. O MQTT-SN foi projetado de forma que seja independente dos serviços de rede subjacentes. Qualquer rede que forneça um serviço de transferência de dados bidirecional entre qualquer nó e um determinado (um *gateway*) deve ser capaz de suportar o MQTT-SN.

2.4 CONTIKI

Para o desenvolvimento do projeto foi necessário usar um software com disponibilidade. Contiki é um sistema operacional de código aberto para a Internet das Coisas, conecta pequenos microcontroladores de baixo custo e baixo consumo de energia à internet.

A Contiki fornece uma poderosa comunicação da internet de baixo consumo, suporta IPv6 e IPv4, juntamente com os recentes padrões de baixo consumo como *IPv6 over Low power Wireless Personal Area Networks (6LoWPAN)*, e até mesmo roteadores sem fio podem ser operados por bateria.

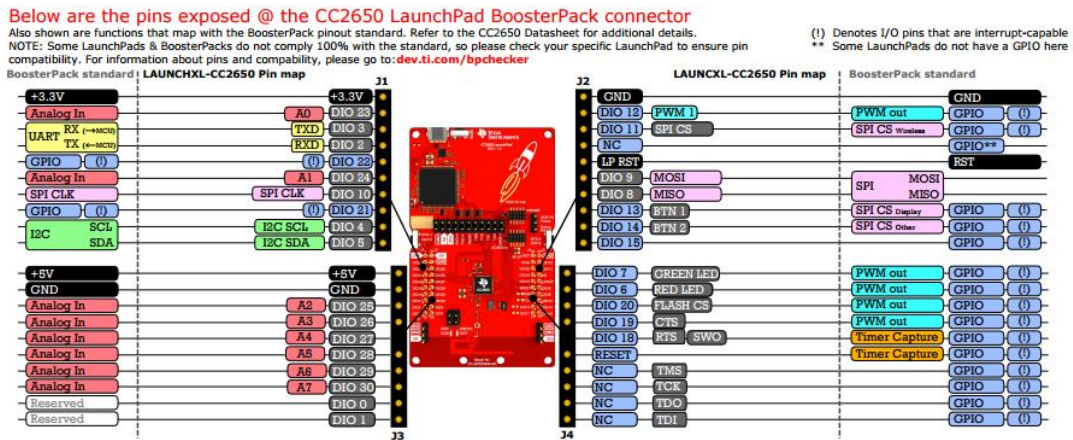
Com o Contiki, o desenvolvimento é fácil e rápido. Os aplicativos Contiki são gravados no padrão C, com o simulador Cooja. As redes Contiki podem ser emuladas antes de serem gravadas no hardware e o Instant Contiki fornece um ambiente de desenvolvimento inteiro em uma única descarga. O logo Contiki foi apresentado na Figura 5.

Figura 5 - Logo do Contiki



Fonte: Kumar (2016).

Figura 7 - Portas do controlador CC2650



Fonte: Braddy (2017).

Para nosso projeto vamos a usar os pines análogos que vão desde A0 a A7, os quais são para inserir dados lidos por os sensores. Para a saída de dados usamos os pines DIO que vão desde DIO 0 a DIO 30 para a saída de dados.

3.2 SOFTWARE

Para nosso projeto foi utilizado o software livre Contiki os que é um sistema operativo “Open Source” que funciona para projetos de Internet das Coisas, ele permite a pequenos sistemas de baixa potência ter comunicação com internet, é muito útil para sistemas de monitoramento, controle industrial e automação.

Se pode usar com todo tipo de protocolos de internet, como IPv4, IPv6, 6LoWPAN, entre outros.

Nosso projeto usa o *Integrated Development Environment* (IDE) da placa CC2650 de Texas Instruments “Code Composer Studio”, ele faz parte de kit da placa quando você compra. O logo de *Code Composer* esta na Figura 8.

Figura 8 - Logo do Code Composer



Fonte: Autoria própria¹.

3.3 CONFIGURAÇÃO DA REDE

O presente projeto vai implementar um sistema de irrigação, no qual o *broker* vai fornecer os estados *publish* e *subscribe* dos sensores que estarão espalhados pelos cultivos, A solução será capaz de analisar variáveis ambientais, do local em que o cultivo está inserida, lendo sensores espalhados pela propriedade agrícola e, quando estes tiveram alguma informação eles enviam. O servidor 6lbr vai fornecer endereços IPv6 que e necessário para fazer comunicação nas redes MQTT-SN, estes endereços chegam a nossa rede por meio de cabo de rede par trançado com acesso a internet que vai chegar a nosso *bourber router* que faz o trabalho de *broker*.

3.3.1 Broker MQTT-SN

No Border Router tem-se um Raspberry pi que faz o trabalho de receber os endereços IPv6 de nosso servidor, depois um hardware para realizar a interface entre Border Router rodando no Linux e a rede 802.15.4. que é o protocolo de transmissão, a principal ferramenta de transmissão e o microcontrolador CC2650 da Texas Instrument que e uma das placas que trabalha com IoT (*Internet of Things*, ou Internet das Coisas).

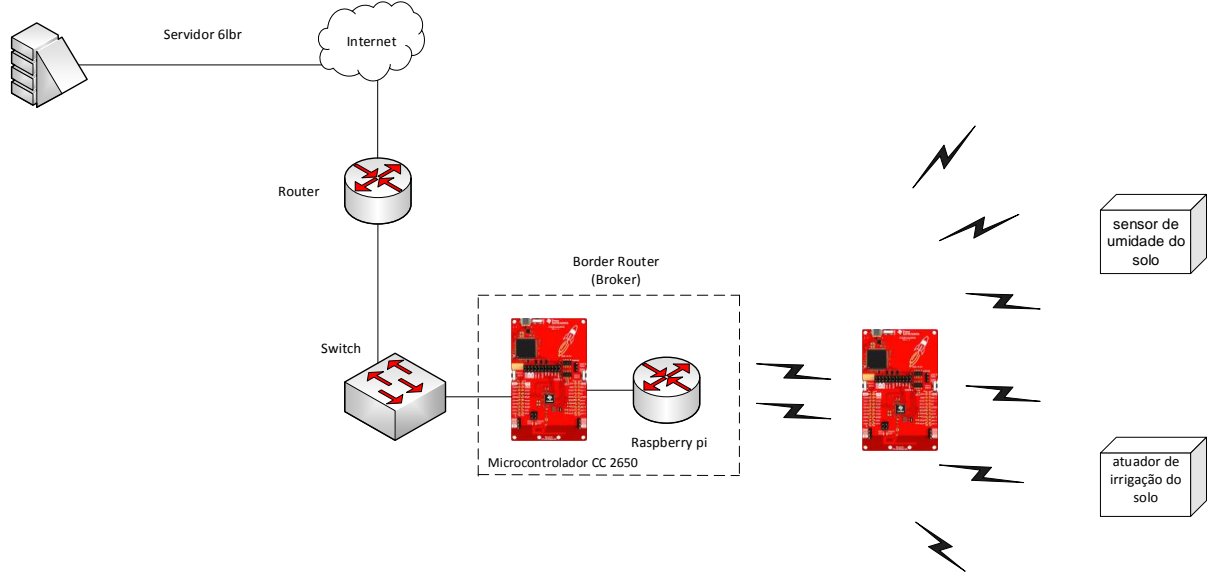
6LoWPAN (IPv6 em redes de área pessoal sem fio de baixa potência) é um padrão que permite o uso do IPv6 em redes baseadas no padrão IEEE 802.15.4.

¹ Imagem disponível em: <https://en.wikipedia.org/wiki/Code_Composer_Studio>. Acesso em: 24 out. 2018.

Isso possibilita que dispositivos como os nós de uma rede sem fio se comuniquem diretamente com outros dispositivos IP.

O desenho usa um servidor 6lbr para fornecer os endereços IPv6 como já se indicou, e por meio de internet chega-se a ao *router* que conecta a uma rede de área local com um *switch* para o *Broker*, como apresentado na Figura 9.

Figura 9 - Diagrama do sistema



Fonte: Autoria própria.

A solução tem que se adaptar facilmente a inclusão ou exclusão dos nós dos sensores e atuadores da solução já em execução, modulando de acordo como o nível de controle desejável pelo agricultor e viabilidade tecnológica do ponto de vista de comunicação, vai ser possível adicionar novos recursos e funcionalidades à solução para atender diferentes necessidades referentes à gestão do sistema de irrigação.

3.4 SENSORES

3.4.1 Sensor de Umidade do Solo

Este sensor faz a leitura da umidade do solo em que está inserido e transmite o valor da umidade lida para outros sensores que fazem uso desta para tomar decisões no acionamento da irrigação autônoma e geração de base histórica para relatórios. A leitura dos valores de umidade transmitidos do solo varia de

acordo com o estado atual da bomba de irrigação (a qual pode estar ligada ou desligada); o qual trabalha conjunto ao sensor de umidade.

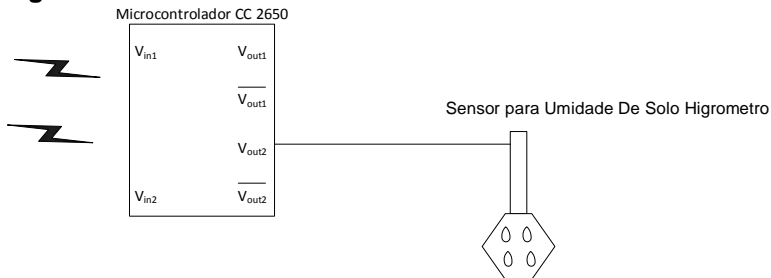
Quando a bomba de irrigação esta detectando umidade do solo, ele envia uma sinal ao atuador de irrigação para que comece a funcionar, o qual trabalha conjunto ao sensor de umidade.

Este bloco funcional tem os seguintes itens:

- Um microcontrolador CC2650 onde o sistema embarcado programado é executado para realizar as tarefas designadas;
- Um Sensor de umidade de solo higrômetro que é responsável pelo monitoramento da umidade do solo em que está inserido, o sensor usa a bateria de 5v da placa do microcontrolador CC2650, por isso não precise de fonte externa.

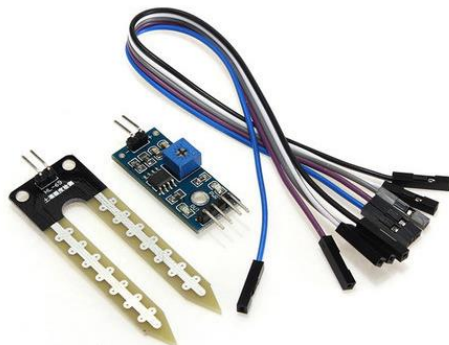
Pode-se observar a configuração do sensor de umidade do solo na Figura 10 e o sensor de umidade que foi utilizado na Figura 11.

Figura 10 - Sensor de umidade do solo



Fonte: Aatoria própria.

Figura 11 - Shield sensor de umidade do solo



Fonte: Aatoria própria².

² Imagem disponível em: <<https://www.betacomercial.com.br/loja/produto/shield-sensor-umidade-solo-higrometro>>. Acesso em: 24 out. 2018.

3.4.2 Atuador de Irrigação do Solo

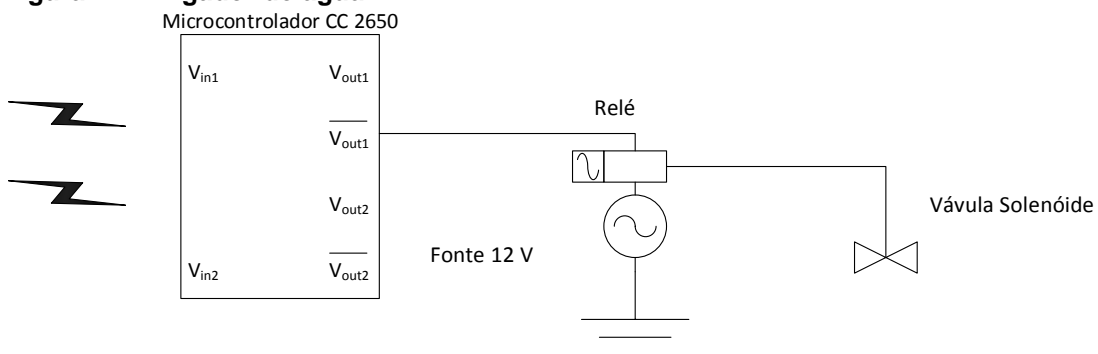
O atuador é responsável de espalhar a água se for necessário em circunstâncias de acordo aos parâmetros de umidade, ele trabalha conjunto ao sensor de umidade, o qual fornece informação sobre o solo se precisa de irrigação. Os dados predicamente configurados e adequados ao nó do cultivo o qual é designado como responsável, enviam uma mensagem do sensor que percebeu a falta de umidade com a informação necessária ao irrigador, o qual recebe esse parâmetro de necessidade de umidade do cultivo para começar a irrigar.

Este bloco funcional será composto pelos seguintes itens:

- Um microcontrolador CC2650 onde o sistema embarcado programado é executado para realizar as tarefas a ele designadas;
- Um relé 5 V de 1 canal para abrir e fechar o circuito que irá acionar a bomba válvula;
- O relé é alimentado por uma bateria externa de 12v, e uma pilha recarregável;
- Uma válvula solenoide que permite ou impede o fluxo de água nas mangueiras de irrigação;
- uma mangueira que está ligada à fonte de água para irrigar.

Na Figura 12 é apresentada a configuração da válvula solenoide e na Figura 13 a válvula utilizada na implementação do projeto.

Figura 12 - Irrigador de água



Fonte: Autoria própria.

Figura 13 - Shield válvula solenoide água 12V/ 3/4



Fonte: Autoria própria³.

³ Imagem disponível em: <<https://www.betacomercial.com.br/loja/produto/shield-valvula-solenoide-agua-12v-3-4>>. Acesso em: 24 out. 2018.

4 DESENVOLVIMENTO

MQTT-SN não requer um protocolo orientado à conexão (como o TCP), ele usa CONNECT para determinar os clientes conectados, dormindo ou ativos. Deve enviar uma mensagem *keep alive* a cada *keep alive duration* que é o tempo de resposta da mensagem *keep alive* que foi indicado na mensagem de CONNECT.

Para fechar usa DISCONNECT com *sleep duration*, onde o cliente passa para o estado ASLEEP, mas se um cliente não manda mensagens após *sleep duration*, este passará para o estado LOST. Durante o estado de ASLEEP, todas as mensagens são armazenadas no *gateway* para serem enviadas após o nó acordar.

Toda a troca da mensagens acontece nos tópicos que estão implementados em cada microcontrolador espalhado em diferentes áreas do cultivo. Cada nó tem um bloco de sensor de umidade de solo e atuador de irrigação de solo, que trabalhar em conjunto para fornecer uma resposta à necessidade dos cultivos.

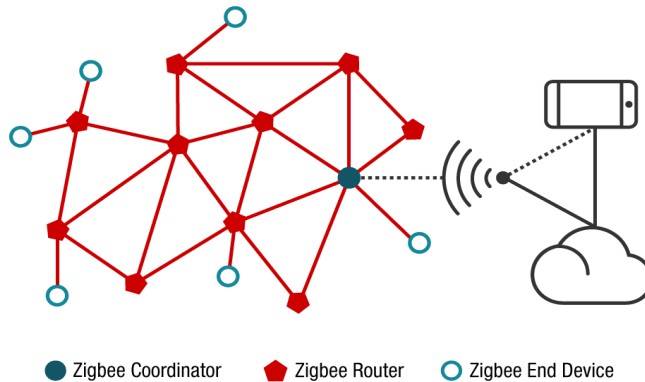
4.1 ZIGBEE

Zigbee é um conjunto de especificações para a comunicação de redes de sensores sem fio, tem ênfase na baixa potência de operação, na baixa taxa de transmissão de dados e no baixo custo de implementação. Usa como padrão IEEE 802.15.4 que funciona com o modelo OSI.

Com um alcance reduzido de não mais de 100 metros a comunicação entre duas unidades poder ser repetida sucessivamente pelas unidades existentes na rede até atingir o destino final.

Os pontos da rede podem funcionar como retransmissores de informação, usando uma malha (Mesh) de unidades Zigbee que pode ser implementada e uma extensão doméstica ou industrial sem necessidade de utilizar ligações elétricas entre elas. Funciona em modo de rede transparente para que todas possam receber dados, e no modo criptografado para segurança. Pode-se observar uma rede Mesh em Zigbee na Figura 14.

Figura 14 - Rede mesh em Zigbee



Fonte: Zigbee (2018).

4.2 IEEE 802.15.4

IEEE 802.15.4 é um padrão técnico que define a operação de redes sem fio de baixa taxa (LR-WPANs). É a base para as especificações Zigbee, cada uma das quais estende ainda mais o padrão, desenvolvendo as camadas superiores que não estão definidas no IEEE 802.15.4. Alternativamente, ele pode ser usado pela 6LoWPAN, a tecnologia usada para entregar a versão IPv6.

4.3 ORGANIZAÇÃO DOS TÓPICOS PUBLICADOS

Os sensores estão espalhados nas áreas do cultivo em uma fazenda, como pode-se ver na Figura 15.

Figura 15 - Área do cultivo



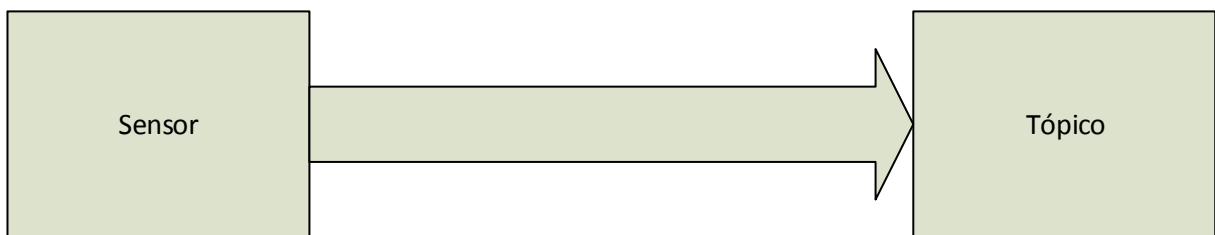
Fonte: Autoria própria.

Por exemplo, tem-se uma área fora de uma fazenda, onde se tem dois sensores e dois atuadores em pares de ambas funções tanto do sensor de umidade de solo higrômetro, quanto do atuador de irrigação do solo, porque trabalham em conjunto, não se pode perceber nenhuma conexão só cada bloco com seu sensor e atuador respectivo.

As mensagens “will and testament”, vão fornecer informação ao *broker* sobre o estado do sensor com a data e hora desse momento, no exemplo pode-se ver a resposta de uma desconexão que aconteceu em um sensor, então esse envia uma mensagem, com a informação apresentada na Figura 16.

Figura 16 - Exemplo de um tópico

Will and Testament



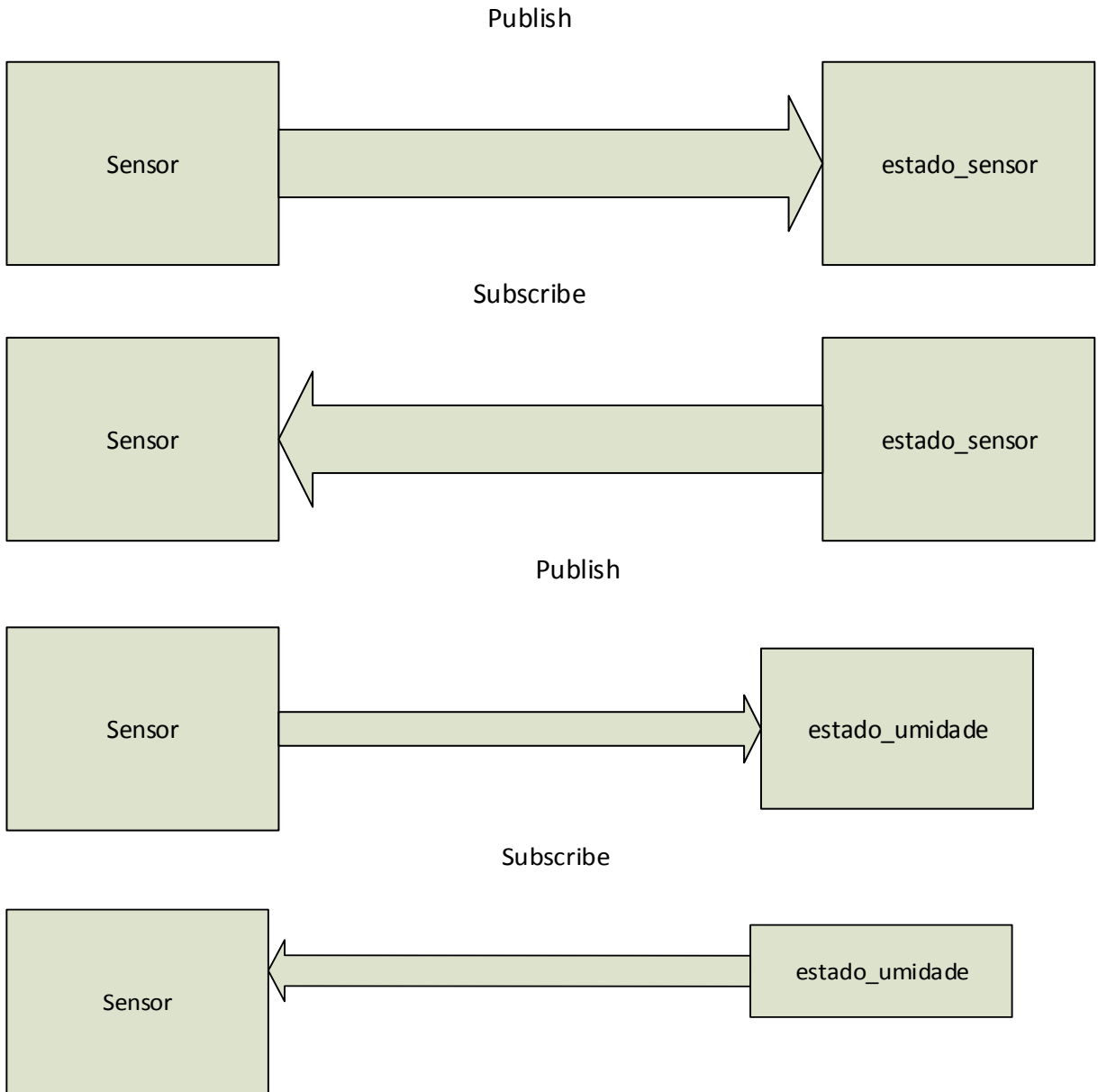
“Valor lido:12 ”

Fonte: Autoria própria.

Os tópicos “ estado_sensor ” e “ estado_umidade ” mostram todo o conteúdo que tem uma troca de mensagens, o estado_sensor mostra o valor lido pela válvula solenoide sobre nível de umidade do solo, o estado_umidade mostra a resposta deste e começa o procedimento de irrigação com os valores de umidade do solo fornecidos pelo sensor de umidade. Os tópicos são os seguintes:

- estado_umidade: Acionamento da Válvula Solenoide (Irrigação parada, Pouca irrigação ou Irrigação Completa);
- estado_sensor: Valor lido (0 a 1020).

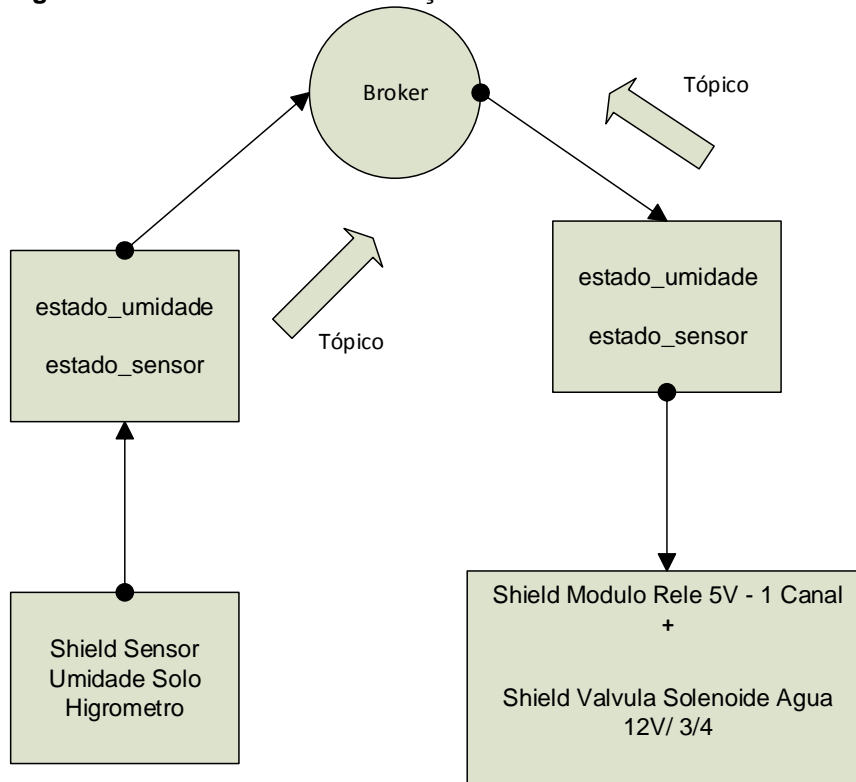
A troca de mensagens dos tópicos pode ser observada na Figura 17.

Figura 17 - Troca de mensagens

Fonte: Autoria própria.

Os tópicos da topologia desenhada podem ser observados na Figura 18.

Figura 18 - Sistema de comunicação



Fonte: Autoria própria.

Por cada sensor de umidade de solo ou higrômetro se envia as mensagens, que é o sensor que percebe os dados da terra, e envia para o *broker* para que elas possam acionar o *shield* válvula solenoide, que funciona com um rele de 5 V este possa começar a espalhar água.

Desse modo, se tem um lugar no cultivo um N número de sensores, onde:

- Código da área onde está o sensor: código U001 (umidade);
- Número de sensores no lugar: de H001 à H020 (higrômetro).

Se tem então:

U001/H001/estado_umidade
 U001/H001/estado_sensor1
 U001/H002/estado_umidade
 U001/H002/estado_sensor2

4.3.1 Formato das Mensagens

A forma geral de uma mensagem MQTT-SN pode ser observada na Figura 19.

Figura 19 - Mensagem MQTT-SN

Message Header (2 or 4 octets)	Message Variable Part (n octets)
-----------------------------------	-------------------------------------

Fonte: Stanford-Clark e Truong (2013, p. 6).

Com um cabeçalho (*Message Header*) de 2 ou 4 octetos e uma parte da mensagem de tamanho variável (*Message Variable Part*). Embora o cabeçalho sempre presente e contenha campos fixos, a mensagem variável dependem do tipo de conteúdo que se estiver enviando.

4.4 MENSAGEM PUBLISH

Todas as mensagens são descritas com o campo *Length* de 1 octeto. Os formatos das mensagens no caso do campo *Length* de 3 octetos podem ser derivados diretamente e, portanto, não são mencionados. Pode-se ver o tamanho de uma mensagem PUBLISH na Figura 20.

Figura 20 - Mensagem PUBLISH

Length (octet 0)	MsgType (1)	Flags (2)	TopicId (3-4)	MsgId (5-6)	Data (7:n)
---------------------	----------------	--------------	------------------	----------------	---------------

Fonte: Stanford-Clark e Truong (2013, p. 13).

Esta mensagem (Figura 20) é usada por clientes e gateways para publicar dados para um determinado tópico, as partes de *Length* (comprimento) e *MsgType* já foram vistas anteriormente. A seguir são apresentados os flags (sinalizadores) que se tem:

- DUP: indica se a mensagem é enviada pela primeira vez ou não;
- QoS: contém o nível de QoS para esta mensagem PUBLICAR;
- Retenha: contém o Retainflag;
- Will: não usado;

- CleanSession: não usado;
- TopicIdType: indica o tipo do ID do tópicos contido no campo TopicId.

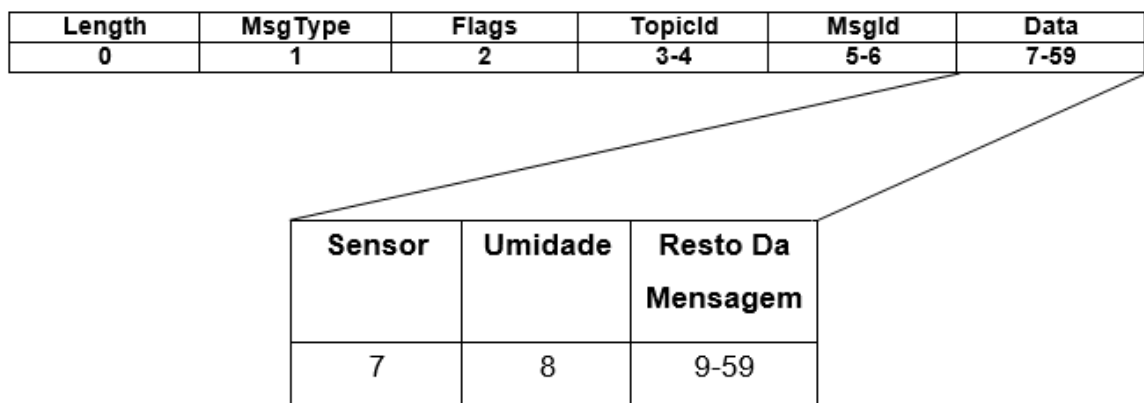
TopicId contém o valor do id do tópicos ou o nome curto do tópicos para o qual os dados são publicados e MsgId tem o mesmo significado que o MQTT “Message ID”; relevante apenas no caso de níveis 1 e 2 de QoS, caso contrário se codifica como 0x0000, e dados que é o espaço onde se desenvolve nossa mensagem.

Agora de acordo com o espaço de dados que temos no campo dados, podemos ver como se distribuem as mensagens em uma rede Zigbee, onde um gateway não precisa ser hospedado por um nó coordenador. No entanto, ele deve residir em um nó de roteador sempre ativo para poder receber mensagens do cliente a qualquer momento, devido à curta duração da carga útil do Zigbee network / APSlayer, o comprimento máximo de uma mensagem MQTT-SN é restrito a 60 octetos. Nossa mensagem fica assim:

Esta configuração permite que depois de começar as publicações, não seria necessário o registro de novos tópicos, porque os tópicos pré-definidos, já foram implementaremos o que permitiria que o sensor já começasse a publicar mensagens logo após sua conexão com o gateway, já que funciona com MQTT-SN. Como exemplo tem-se o tópicos U001/H001/estado_umidade

O uso que tem-se dos campos da mensagem de umidade que é fornecido pelo sensor, pode ser observado na Figura 21.

Figura 21 - Mensagem de umidade



Fonte: Autoria própria.

Assim tem-se o resultado do espaço da mensagem ocupados pela informação da locação do sensor de umidade, com espaço para 30 campos livres para qualquer data na mensagem.

4.5 MENSAGEM SUBSCRIBE

A mensagem SUBSCRIBE (Figura 22) é usada por um cliente para assinar um determinado nome de tópico.

Figura 22 - Mensagem SUBSCRIBE

Length (octet 0)	MsgType (1)	Flags (2)	MsgId (3-4)	TopicName or TopicId (5:n) or (5-6)
---------------------	----------------	--------------	----------------	--

Fonte: Stanford-Clark e Truong (2013, p. 15).

Segundo a imagem da Figura 22, tem-se as seguintes partes:

- Comprimento (*Length*) e *MsgType*.
- Sinalizadores (*Flags*):
 - DUP: igual ao MQTT, indica se a mensagem é enviada pela primeira vez ou não.
 - QoS: igual a MQTT, contém o nível de QoS solicitado para este tópico.
 - Retenção: não utilizada.
 - Vontade: não utilizada.
 - CleanSession: não utilizada.
 - TopicIdType: indica o tipo de informação incluída no final da mensagem, nomeadamente “0b00” nome do tópico, “0b01” ID do tópico pré-definido, “0b10” nome do tópico curto e “0b11” reservado.
- *MsgId*: deve ser codificado de forma que possa ser usado para identificar a mensagem SUBACK correspondente.
- *TopicName* ou *TopicId*: contém o nome do tópico, o ID do tópico ou o nome do tópico curto, conforme indicado no campo *TopicIdType*.

4.6 INTERPRETAÇÃO DOS RESULTADOS

Os dados obtidos são de acordo aos teste feitos como o sensor de umidade higrômetro, os quais aparecem na Tabela1.

Tabela 1 - Funcionamento dos sensores

Valor analógico da leitura do sensor	Umidade do solo	Funcionamento da válvula solenoide
0 -599	Molhado	Irrigação parada
600 – 799	Levemente umido	Pouca irrigação
800 – 1023	Seco	Irrigação completa

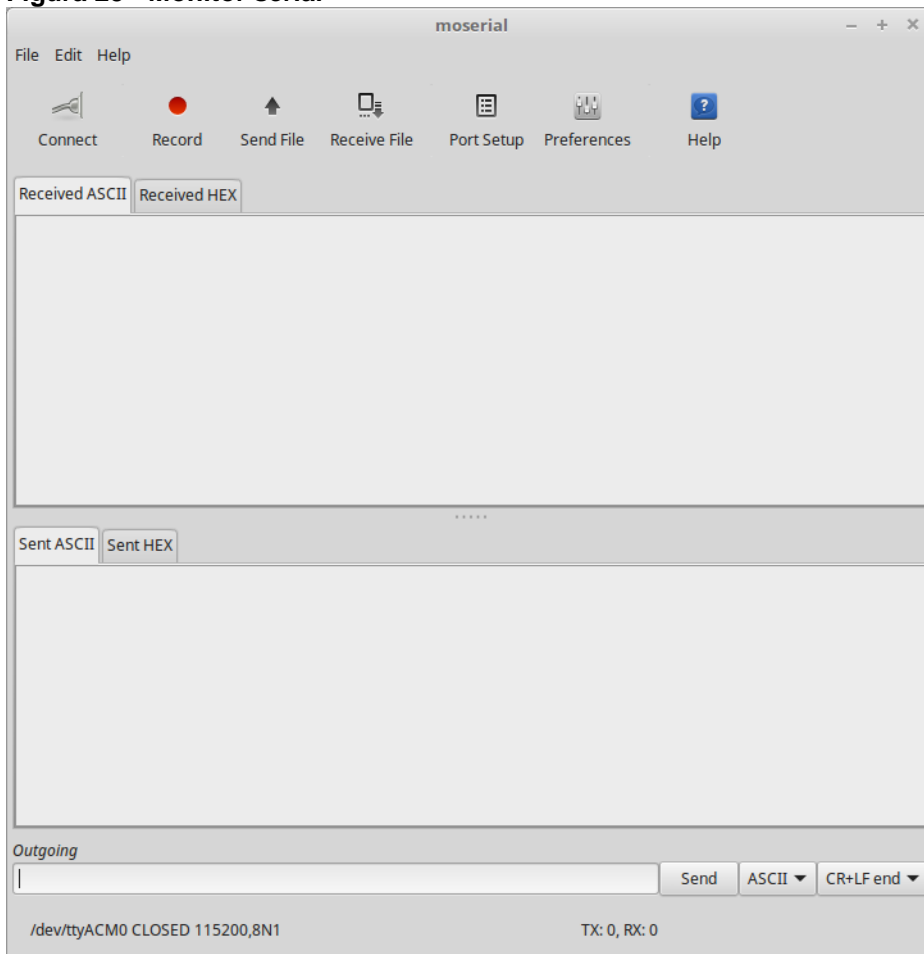
Fonte: Autoria própria.

Para medir a umidade do cultivo usa-se o sensor higrômetro, o qual mede a umidade do solo pela variação da condutividade, os valores obtidos pela saída analógica variam de 0 submerso em água a 1023 no ar ou solo muito seco, um solo levemente úmido daria valores entre 600 a 700. Um solo seco ao redor de 800 a 1023.

A leitura dos dados do sensor de umidade e feita pela porta analógica que existente no microcontrolador, esses dados por meio do código do microcontrolador acionam a válvula solenoide que funciona com um rele de 5 v, a válvula solenoide funciona com uma porta de saída digital que existente no microcontrolador.

4.6.1 Testes

Para a captura de dados se usa Monitor Serial (Figura 23), que permite que você veja toda a atividade iniciada por outros aplicativos em uma porta serial, incluindo abertura, fechamento, alteração de configuração serial, mudança de linha de controle de status e, é claro, os dados reais fluindo pela porta. Atualmente este *plugin* está disponível apenas no Microsoft Windows e Linux, para o projeto foi usado a versão do linux.

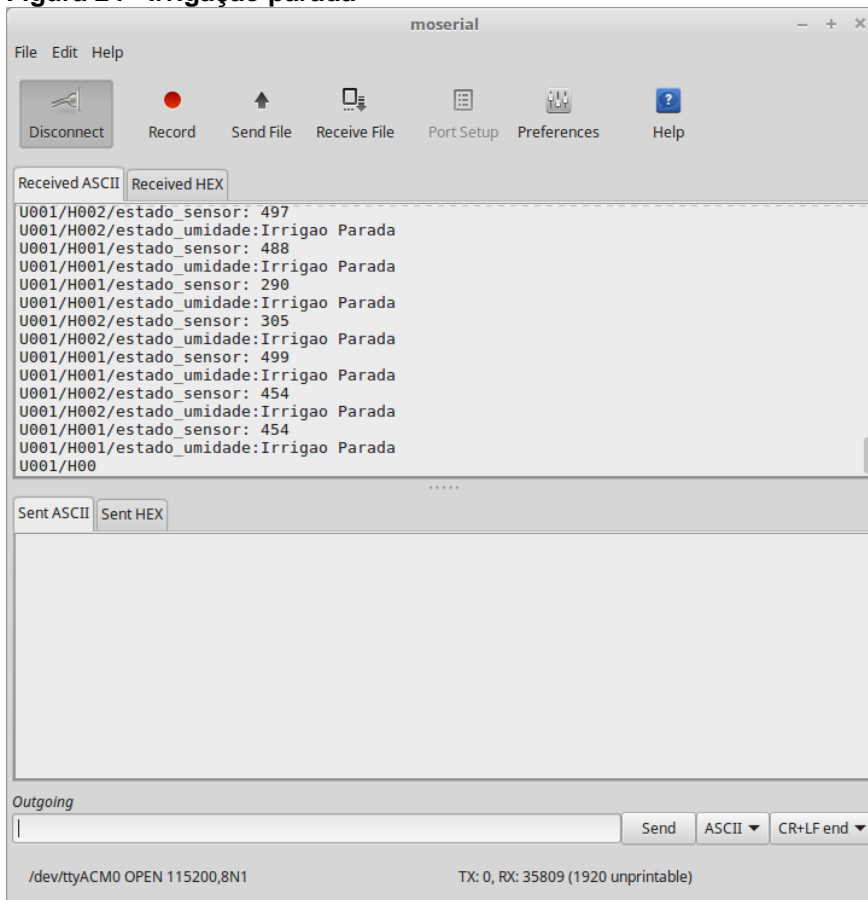
Figura 23 - Monitor serial

Fonte: Autoria própria.

A resposta dos sensores tem três resultados de acordo aos tópicos já mencionados: a) Irrigação parada; b) Pouca irrigação; e c) Irrigação completa.

4.6.1.1 Irrigação parada

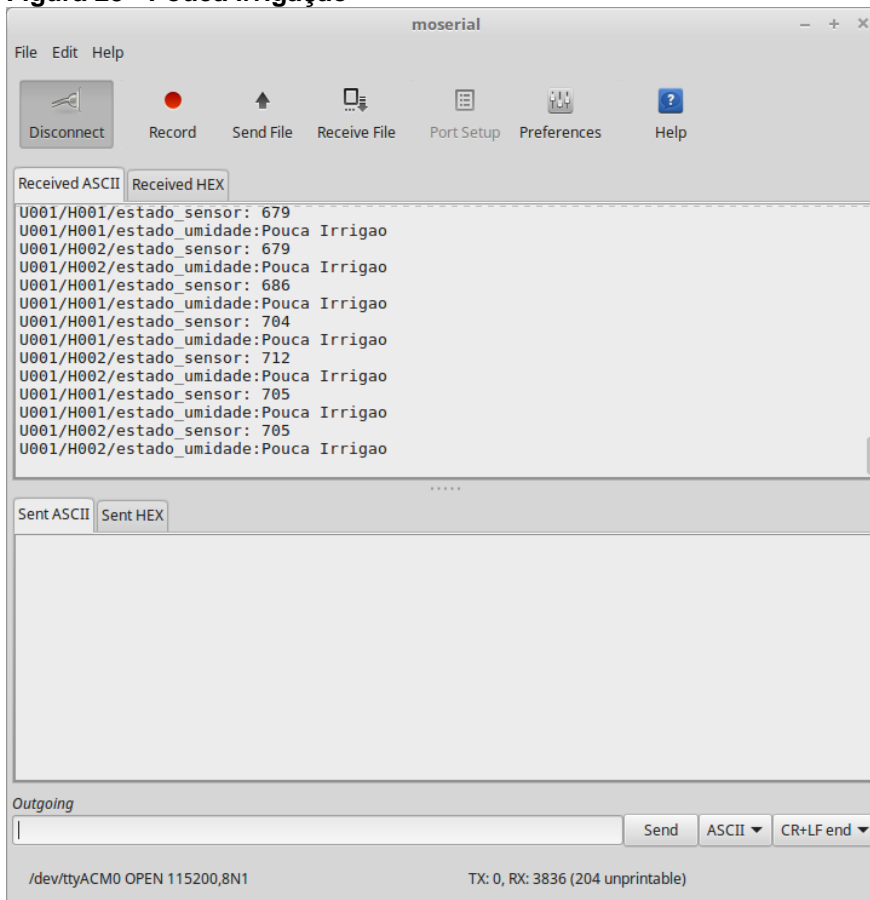
Como se mostrou na Tabela 1, de 0 a 599 no valor do sensor não se tem dados de umidade do solo, por isso o estado do tópico umidade está com irrigação parada, como se pode ver a Figura 24 na resposta do Monitor Serial.

Figura 24 - Irrigação parada

Fonte: Autoria própria.

4.6.1.2 Pouca irrigação

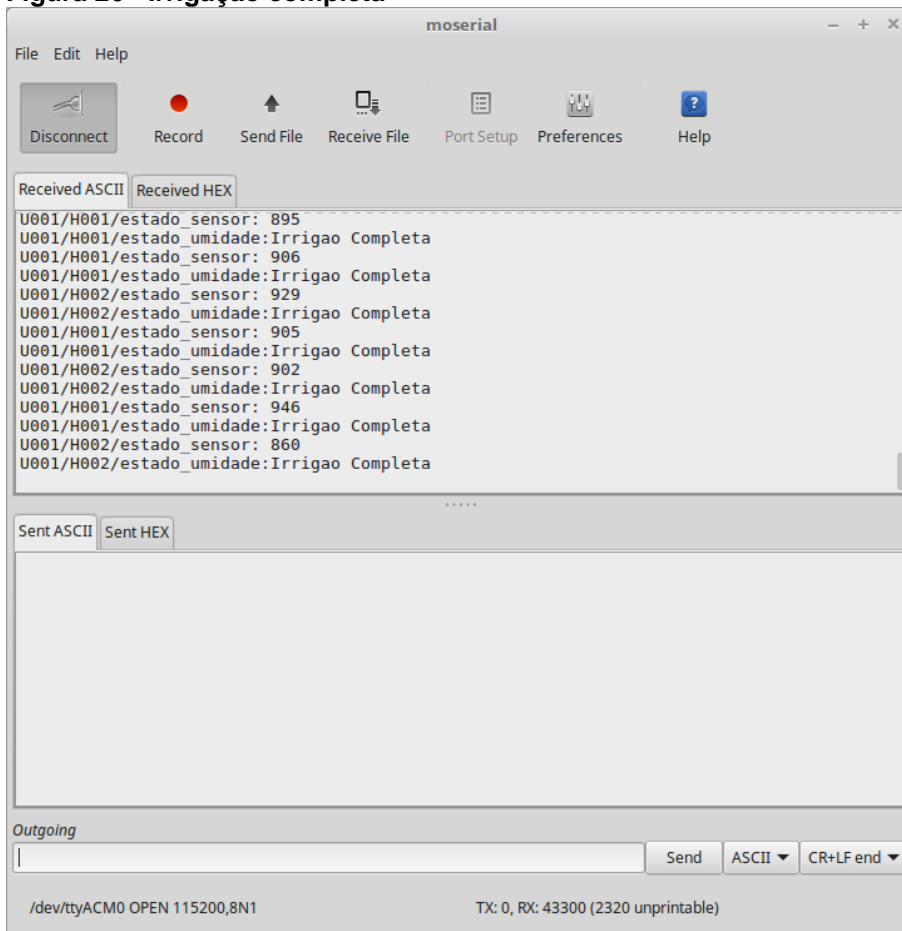
Como se mostrou na Tabela 1, de 600 a 799 no valor do sensor se tem pouca umidade do solo, por isso o estado do tópico umidade está com pouca irrigação, como pode ser observado na Figura 25 na resposta do Monitor Serial.

Figura 25 - Pouca irrigação

Fonte: Autoria própria.

4.6.1.3 Irrigação completa

Como se mostrou na Tabela 1, de 800 a 1023 no valor do sensor se tem solo seco e precisa de irrigação completa, por isso o estado do tópico umidade está com irrigação completa, como pode ser observado na Figura 26 na resposta do Monitor Serial.

Figura 26 - Irrigação completa

Fonte: Autoria própria.

5 CONCLUSÕES

Depois de realizar o seguinte projeto se pode ter a certeza que com a tecnologia de redes de sensores sem fio se pode chegar a diminuir os problemas na área dos cultivos, especialmente em lugares afastados onde não se tem muitas pessoas para trabalhar 24 horas do dia na irrigação do solo e esperando até que o solo fique seco, porque com ajuda de um sensor se pode perceber mais claro a umidade do solo desde dentro sem fazer trabalho com as mãos que pode chegar a ser perigoso, com ajuda de um sistema de irrigação você pode monitorar seu solo sem se arriscar a condições climáticas extremas, se pode chegar à conclusão que em um futuro próximo se pode trabalhar com mais sensores que podam fornecer melhores detalhes do solo e outras características no cultivo.

O controlador CC2650 funcionou de acordo com os resultados previstos, embora se tenha outros microcontroladores com as mesmas características iguais se pode dizer que funciona de acordo ao planejado. Sobre Contiki que é um software novo e quase experimental, precisa-se aprofundar os estudos sobre este, que tem muito futuro com o desenvolvimento das redes de sensores sem fio.

REFERÊNCIAS

BRADDY, Rick. **J-Box Prototype 1 Design and Layout**. Copyright© W5FCX, publicado em: 09 mar. 2017. Disponível em: <<https://www.w5fcx.net/2017/03/09/j-box-prototype-1-design-and-layout/>>. Acesso em: 25 out. 2018.

CC2650. **SimpleLink™ CC2650 wireless MCU LaunchPad™ Development Kit**. Copyright© 1995-2018 Texas Instruments Incorporated. Disponível em: <<http://www.ti.com/tool/LAUNCHXL-CC2650>>. Acesso em: 25 out. 2018.

COLINA, Antonio Liñán; et al. **Internet of Things (IoT) in 5 days**. Versão eletrônica 1.1, publicada em: 24 jun. 2016. Disponível em: <<http://www.iet.unipi.it/c.vallati/files/IoTinFivedays-v1.1.pdf>>. Acesso em: 27 out. 2018.

GLOBO. **Produtores Investem em irrigação para produzir banana**: Região Noroeste é importante área de cultivo da fruta em São Paulo. Copyright© 2000-2018 Globo Comunicação e Participações S.A. Por Nosso Campo, TV TEM, publicado em: 15 jul. 2018. Disponível em: <<https://g1.globo.com/sp/sorocaba-jundiai/nosso-campo/noticia/produtores-investem-em-irrigacao-para-produzir-banana.ghtml>>. Acesso em: 20 out. 2018.

KUMAR, Arun. **Contiki OS vs Windows 10 for Internet of Things**. Copyright© 2018 The Windows Club, publicado em: 30 jan. 2016. Disponível em: <<https://www.thewindowsclub.com/contiki-os-vs-windows-10-internet-things>>. Acesso em: 15 nov. 2018.

MAIO, Henrique Rodrigues; KLING, Hugo Henrique de Melo; ALMDEIRA, Guilherme Magalhães. **Endereçamento do IPv6**. Universidade Federal do Rio de Janeiro, Redes de Computadores II, segundo semestre de 2018. Disponível em: <https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2013_2/ipv6/>. Acesso em: 25 out. 2018.

STANFORD-CLARK, Andy; TRUONG, Hong Linh. **MQTT For Sensor Networks (MQTT-SN)**: Protocol Specification Version 1.2. Copyright© 1999-2013 International Business Machines Corporation (IBM), publicado em: 14 nov. 2013. Disponível em: <1999 – 2013 International Business Machines Corporation (IBM)>. Acesso em: 25 out. 2018.

ZIGBEE. **SimpleLink™ Zigbee wireless MCUs**. Copyright© 1995-2018 Texas Instruments Incorporated. Disponível em: <<http://www.ti.com/wireless-connectivity/simplelink-solutions/zigbee/overview.html>>. Acesso em: 25 out. 2018.