

Gateway LoRaWAN com Conexão OPC UA para Dispositivos IoT Industriais

Angelo Guimarães Silva * Hugo César Coelho Michel **

* *Concert Cloud, Rua Antônio de Albuquerque, 759, Belo Horizonte, MG, Brasil (e-mail: angelo.guimaraes@concert.com.br)*

** *Dep. de Engenharia Eletrônica, Escola de Engenharia da Universidade Federal de Minas Gerais, Av. Pres. Antônio Carlos, 6627 - Pampulha, Belo Horizonte, Minas Gerais, Brasil (e-mail: hugomichel@ufmg.br)*

Abstract: This article presents the integration of a long-range wireless sensor network, using LoRa technology, the LoRaWAN protocol and an OPC TCP network with the OPC UA standard. This proposal allows the creation of cyber-physical spaces where all elements can communicate in a secure, adaptable and scalable way, offering a IIoT (Industrial Internet of Things) connectivity solution, useful for Industry 4.0. Due to the specificities of each protocol (LoRa, LoRaWAN and OPC UA), developments in hardware, firmware and software are presented with the objective of obtaining an integrated solution capable of providing data from a LoRaWAN network directly to an HMI, using the OPC UA standard. Details of the embedded electronic system developed for the LoRa sensor nodes are presented. The implementation steps of the LoRaWAN - OPC UA Gateway, based on the OpenWrt system are also presented. Finally, the test results of the developed LoRa sensor nodes are discussed, as well as the integration with a SCADA operation interface, demonstrating the feasibility of the proposed integration.

Resumo: Este artigo apresenta a integração de uma rede de sensores sem fio de longo alcance, utilizando a tecnologia LoRa, o protocolo LoRaWAN e uma rede OPC TCP com padrão OPC UA. Esta proposta permite a criação de espaços ciberfísicos onde todos os elementos podem se comunicar de forma segura, adaptável e escalável, oferecendo uma solução de conectividade IIoT (*Industrial Internet of Things*), útil para a Indústria 4.0. Devido às especificidades de cada protocolo (LoRa, LoRaWAN e OPC UA), desenvolvimentos em hardware, firmware e software são apresentados com o objetivo de obter uma solução integrada capaz de fornecer dados de uma rede LoRaWAN diretamente para um IHM, usando o OPC UA. Detalhes do sistema eletrônico embarcado desenvolvido para os nós sensores LoRa são apresentados. As etapas de implementação do LoRaWAN - OPC UA Gateway, baseado no sistema OpenWrt também são apresentados. Por fim, são discutidos os resultados de testes dos nós sensores LoRa desenvolvidos, bem como a integração com uma interface de operação SCADA, demonstrando a viabilidade da integração proposta.

Keywords: IIoT, OPC UA, LoRaWAN, OpenWrt, Industry 4.0.

Palavras-chaves: IIoT, OPC UA, LoRaWAN, OpenWrt, Indústria 4.0.

1. INTRODUÇÃO

O uso de tecnologias que visem o conceito de *Internet of Things* (IoT) é um campo vasto a ser explorado, trazendo viabilidade para a Indústria 4.0. O conceito de IoT foi originalmente empregado por Kevin Ashton em 1999, ao propor a ideia de objetos diversos sentindo o mundo ao redor por meio de redes (Rangel, 2014). No momento atual, o IoT pode ser melhor interpretado como interconexões digitais entre objetos, ativos e sistemas antes isolados, mas que agora conseguem se conectar e trocar informações entre si. No contexto industrial surge um novo termo derivado do IoT: o Industrial Internet of Things (IIoT), que leva em conta as necessidades específicas do ambiente industrial no uso de soluções pautadas no conceito de IoT, como segurança, disponibilidade, coleta não intrusiva

de dados no processo industrial e compatibilidade de protocolos de comunicação.

Existem diversos protocolos e tecnologias de comunicação baseados em Internet das Coisas, entre eles tem-se o protocolo LoRaWAN (*Long Range Wide Area Network*) e a tecnologia de rádio transmissão LoRa (*Long Range*). A tecnologia LoRa tem ganhado destaque entre os desenvolvedores por reunir em um único hardware, pequeno e barato, características presentes em redes de sensores aplicadas a IoT e IIoT, como longo alcance, baixo consumo de energia e alta imunidade a ruídos (Compology, 2017; Pursche, 2016). O protocolo LoRaWAN é mantido pela LoRa Alliance e define em sua estrutura de rede uma arquitetura em topologia estrela protegida por criptografia de 128 bits, permitindo a comunicação de dispositivos finais

com um ponto concentrador da rede, o gateway (LoRa Alliance, 2015).

2. TECNOLOGIAS LORA, LORAWAN E OPCUA E SUAS VANTAGENS PARA IIOT

A tecnologia LoRa faz uso da já conhecida faixa de frequências não licenciadas, ISM (*Industrial Scientific and Medical*). A LoRa Alliance define parâmetros regionais em todo o mundo, para que as redes LoRaWAN possam operar sem infringir normas regulamentadoras em qualquer lugar no globo.

A tecnologia de rádio transmissão LoRa, juntamente com o protocolo LoRaWAN, se destaca pela modulação utilizada e pelo modo de operação, capaz de garantir boa relação entre consumo energético e alcance da transmissão, além da alta imunidade a ruídos, ideal para ambientes industriais. LoRa consiste em uma técnica de modulação derivada da tecnologia CSS (*Chirp Spread Spectrum*), que consiste no breve aumento (up-chirp) ou decaimento (down-chirp) da frequência ao longo do tempo, representando os bits do pacote em uma rádio transmissão digital. Essa tecnologia não é novidade, sendo utilizada há longa data em aplicações militares que demandam segurança (Semtech 2015).

Em (Talla et al., 2017), uma análise de frequência de um frame LoRaWAN é apresentada com detalhes de *Preamble*, *Sync*, (up and down)-chirp plus Header and *Payload* com *Error Check*. A codificação do pacote envolve o uso de mais três fatores: **Band width (BW)**, **Code rate (CR)** e **Spread factor (SF)**, conforme (Semtech Corporation, 2015; LoRa Alliance, 2018).

As características da modulação LoRa apresentadas reduzem drasticamente o risco de colisão dos dados, pois é necessário que dois ou mais pacotes cheguem simultaneamente no receptor, com a mesma frequência, mesmo spread factor e mesma largura de banda.

2.1 Protocolo LoRaWAN

LoRaWAN é um protocolo sem fio para redes LPWAN (*Low Power Wide Area Networks*), desenvolvido para uso com transceptores LoRa. O alcance do sinal pode ultrapassar 12 km, porém muitos fatores podem influenciar nesse alcance, como obstáculos, qualidade da antena, além das configurações da rede.

A tecnologia LoRa atua na camada física, primeira camada do modelo OSI (*Open System Interconnection*). Já o protocolo LoRaWAN atua na segunda camada, MAC (*Media Access Control*) (LoRa Alliance, 2015). O protocolo LoRaWAN define as características dos nós finais da rede conforme três classificações: **Classe A** (transmissão assíncrona e o menor consumo energético), **Classe B** (transmissão síncrona com janelas de recepção agendadas) ou **Classe C** (transmissão síncrona com janela de recepção aberta indefinidamente) (LoRa Alliance, 2015).

Ainda é importante dizer que os nós da rede e o servidor possuem em seus registros as chaves *Device Address* (DevAddr), *Device Extended Unique Identifier* (DevEUI), *Application Session Key* (AppSKey), *Network Session Key* (NwkSKey), *Application Key* (AppKey) e *Application Extended Unique Identifier* (AppEUI) usadas para adicionar

segurança aos pacotes e endereçar os nós, conforme detalhado em (LoRa Alliance, 2017).

2.2 Protocolo OPC para arquiteturas diversas

O protocolo OPC (*Open Platform Communications*) visa padronizar a comunicação e interoperabilidade entre os elementos do chão de fábrica e demais níveis da pirâmide de automação, por meio de uma estrutura organizacional de dados (OPC Foundation, 2017; Mahnke et al., 2009), normalmente acessados por uma rede utilizando a pilha de protocolos TCP/IP.

O OPC UA tem sido cotado como o protocolo ideal para aplicações do conceito de Indústria 4.0, visto que ele possibilita a total integração entre os diversos níveis de uma fábrica com o mínimo de modificações na Arquitetura de Controle e Automação (ACA) (OPC Foundation, 2017; Mahnke et al., 2009). O fato de ser um protocolo aberto e independente de plataforma reforça essa preferência. O conceito da arquitetura unificada do OPC UA prevê uma interação mútua entre clientes e servidores mediante a interconectividade entre ambos. Dessa forma, uma mesma ACA pode ter, ao mesmo tempo, diversos servidores e clientes. Um cliente pode estar conectado a vários servidores e um servidor pode estar conectado a vários clientes. É possível, ainda, haver uma instância híbrida de cliente e servidor, útil para facilitar a interface entre diferentes sistemas (OPC Foundation, 2017; Mahnke et al., 2009).

Por fazer uso de uma arquitetura cliente/servidor, o padrão OPC UA também implementa métodos de descoberta em rede, facilitando desse modo a localização de servidores por parte dos clientes e, em seguida, o estabelecimento da sessão entre ambos. Segundo Mahnke et al. (2009) a conexão pode ocorrer sob duas formas: via **OPC TCP**, ou **SOAP HTTP(S)**.

Por possuir três camadas de segurança, que podem ser implementadas conjuntamente ou separadamente, o OPC UA mostra-se como um protocolo bastante seguro. A segurança no padrão OPC UA, consiste em em três etapas (Mahnke et al., 2009):

- **Etapa 1:** utilização de certificados de segurança entre clientes e servidores;
- **Etapa 2:** utilização de mecanismos de autenticação de usuário no servidor;
- **Etapa 3:** criação de uma sessão com dados criptografados na rede.

Tratar a informação apenas como um valor numérico é uma maneira insuficiente para lidar com a imensa quantidade de dados na indústria moderna Commserver (2010). Em consonância, o protocolo OPC UA instituiu uma nova forma de abstrair e modelar dados em rede, sob forma de objetos, *i.e.*, instâncias dotadas de *variáveis, métodos e eventos*.

O conjunto desses objetos, juntamente com seus componentes, constitui o *Address Space*, *i.e.*, uma região de memória no servidor utilizada pelos clientes para acessar as funcionalidades que os objetos fornecem. Todos os objetos no *Address Space* são sinalizados como **nós**, *i.e.*, formas de organizar os objetos como instâncias modeladas por duas características (OPC Foundation, 2017; Mahnke et al.,

2009; Unified Automation, 2020): **Atributos** (elementos que descrevem as características dos nós) e **Referência** (informa como um nó está ligado a outros nós).

3. ARQUITETURA PARA NÓS SENSORES E GATEWAY LORAWAN/OPC UA

A Figura 1 apresenta a arquitetura da proposta visando a integração das características de longo alcance, baixo consumo, baixa taxa de colisão e alta imunidade a ruídos da tecnologia LoRa e LoRaWAN, com o modelo de objetos e conectividade do protocolo OPC UA. A camada física e de enlace do modelo OSI são representadas pelos nós sensores desenvolvidos neste trabalho. Os dados coletados trafegam, inicialmente, pela rede LoRaWAN com destino a um servidor OPC UA, onde clientes OPC UA podem se conectar para coletar os dados provenientes dos nós LoRa. Na camada de controle de acesso, predomina a atuação do protocolo LoRaWAN, dependendo diretamente da ação mútua entre os nós sensores e o *gateway*. Nela, os nós constroem o pacote LoRaWAN e o transmite na rede wireless modulada em LoRa. Por sua vez, o *gateway* age recebendo esse pacote com o objetivo de encapsulá-lo em um pacote UDP/IP com destino a redes externas.

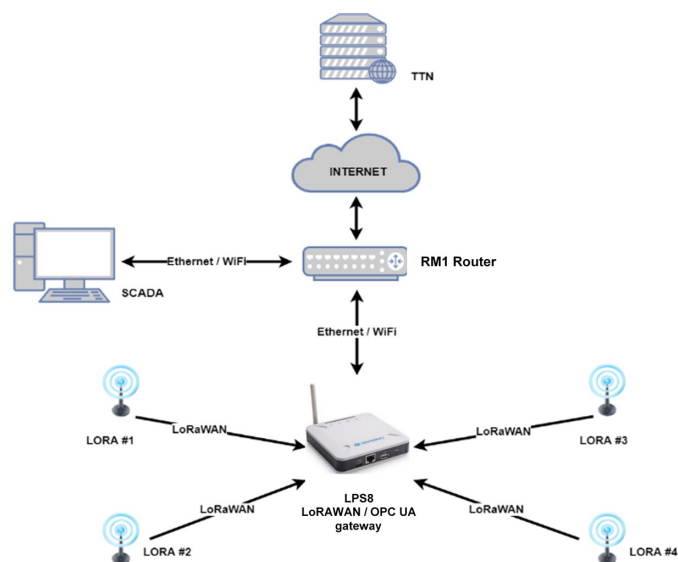


Figura 1. Arquitetura da solução utilizando o *gateway* LoRaWAN/OPC UA.

Dessa forma, uma modificação deve ser feita no *firmware* encaminhador de pacotes LoRaWAN em execução no *gateway*, para permitir que os pacotes recebidos possam ser encaminhados também para a aplicação do servidor OPC UA, que deve estar em execução localmente pelo sistema operacional do *gateway*. Por fim, o nível de aplicação é representado pelos diferentes destinos que os dados podem tomar de acordo com o serviço de IoT configurado no *gateway* e também pelo cliente OPC UA.

A Figura 1 exhibe os principais elementos que compõem toda a solução proposta, com destaque para os equipamentos: o Roteador RM1, a estação executando uma interface SCADA, serviços IoT em nuvem; o *gateway* LPS8 (LoRaWAN/OPC UA), cujo *firmware* é modificado e, finalmente, os nós sensores LoRa, cujos *firmware* e hardware, respectivamente, são todos desenvolvidos neste trabalho.

3.1 Concepção do gateway LoRaWAN/OPC UA

É importante que o *gateway* seja capaz de disponibilizar o conteúdo do *payload* LoRaWAN em um servidor OPC UA. Para alcançar este objetivo é necessário realizar modificações no *firmware* original que roda embarcado no *gateway* e, com isso, obter a integração desejada dos dois protocolos. O *gateway* utilizado é o modelo Dragino LPS8 915MHz fabricado pela Dragino Technology (Dragino, 2017), apresentado na Figura 1.

O sistema operacional (OS) em execução no LPS8 é o OpenWRT 18.06, *i.e.*, um OS kernel GNU/Linux, independente e flexível para sistemas embarcados, especialmente para roteadores (Wich, 2020). O sistema operacional no LPS8 além de executar os algoritmos de roteamento mencionados anteriormente, permite também a execução de softwares escritos em linguagem C, utilizada no desenvolvimento deste trabalho do software capaz de integrar os protocolos LoRaWAN e OPC UA.

O servidor OPC UA é desenvolvido com base na biblioteca open source Open62541, que possui o diferencial de poder ser portada para distribuições de diferentes sistemas operacionais como Microsoft Windows, Linux, QNX e Android (Open62541, 2016). A Open62541 possui funções que permitem que um valor seja disponibilizado como uma variável acessível em um servidor, para tanto é necessário que esse valor seja obtido de alguma fonte. Logo, por meio de um programa escrito em linguagem C, é possível obter valores de diferentes fontes, *e.g.*, um arquivo, ou uma posição de memória, ou uma variável compartilhada com outra aplicação, etc.

A integração proposta faz uso de dados presentes nos pacotes LoRaWAN, que trafegam na aplicação `lora_pkt_fwd`, em execução no *gateway*, a função desta aplicação é a de encaminhar os pacotes da rede LoRaWAN para uma rede UDP/IP. No caso, é necessário desenvolver uma estratégia para enviar esses dados para a aplicação do servidor OPC UA. O programa `lora_pkt_fwd` nativo no LPS8 faz parte do pacote `lora-gateway` desenvolvido pela Semtech. Esse pacote inclui programas utilizados como ponto de partida para o desenvolvimento de um *gateway* multicanal.

Com o código disponibilizado, é possível modificá-lo para permitir a integração com o servidor OPC UA. Portanto, a modificação a ser feita consiste em copiar o pacote LoRaWAN recebido na aplicação `lora_pkt_fwd`, antes dele ser enviado para o servidor padrão via UDP/IP. A cópia do pacote é gravada em um arquivo local que pode ser acessado concomitantemente pela aplicação do servidor.

O servidor OPC UA deve repetir, ciclicamente, a leitura do arquivo compartilhado com a aplicação `lora_pkt_fwd`. Desta forma, sempre que o pacote atual se diferenciar do último pacote lido, a variável atribuída ao endereço presente no pacote atual deve ser atualizada. O endereço do remetente está presente no início do pacote, após o campo `MAC Header`, e o `payload` ao final do pacote, protegido por criptografia AES-CTR. Por fim, a integração desenvolvida ainda depende de um “arquivo de configuração” com os seguintes campos para conhecer os detalhes de cada nó a ser listado no servidor:

- **Endereço:** endereço único de 4 bytes em hexadecimal na 1ª coluna.
- **Descrição:** descrição sobre o nó de até 20 caracteres na 2ª coluna.
- **AppKey:** chave utilizada pelo servidor para descryptografar o *payload* do pacote, deve estar na 3ª coluna.

A Figura 2 apresenta o fluxograma do código para aplicação que dá origem ao *gateway* LoRaWAN - OPC UA, vale lembrar que a aplicação depende que a modificação na aplicação *lora_pkt_fwd* já tenha sido realizado, uma vez que é necessário ter acesso aos pacotes LoRaWAN que estarão disponíveis após a modificação. Ainda na Figura 2 pode-se observar as duas *threads* utilizadas na execução do algoritmo do servidor OPC UA.

À esquerda a *thread* *main()*, onde ocorre: (1) o carregamento das configurações utilizadas no servidor por meio do arquivo de configuração; (2) a verificação da existência do arquivo compartilhado; (3) a criação de uma segunda *thread* e (4) a chamada para execução do servidor OPC UA. À direita, a *thread* *read_data_to_nodes()* executa a leitura continua do arquivo compartilhado, seguida da identificação do endereço do nó de classe A, que deve ter suas respectivas variáveis atualizadas no servidor.

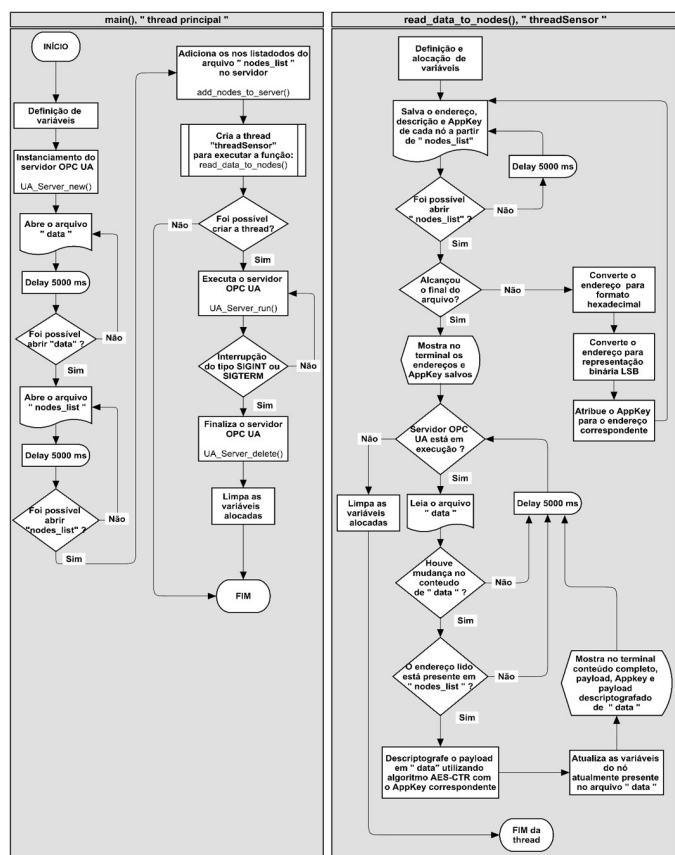


Figura 2. Aplicação LoRaWAN/OPC UA.

3.2 Concepção dos Nós Sensores

Os nós são os pontos onde ocorrem ativamente as medições da planta, os valores lidos nas medições são posteriormente tratados e comprimidos em 4 bytes, para em seguida serem enviados para o serviço configurado no *gateway* e paralelamente enviados para o servidor OPC UA desenvolvido.

Para este trabalho, são desenvolvidas quatro *Print Circuit Boards* (PCBs) para os nós. Em cada PCB há um microcontrolador (μ C) do fabricante Microchip com a função de executar a pilha do protocolo LoRaWAN e manipular os periféricos de I/O (Input/Output).

O projeto eletrônico contempla uma plataforma com um primeiro módulo de tamanho reduzido, onde se encontra o μ C PIC18LF26K22, uma fonte de alimentação e uma interface de I/O. Um segundo módulo, contendo o rádio LoRa, é projetado para se conectar ao primeiro. As Figuras 3a e 3b, apresenta-se as PCBs em estágio de desenvolvimento, bem como após a fabricação. A Figura 3c mostra o módulo de rádio LoRa utilizando o tansceptor SX1276.

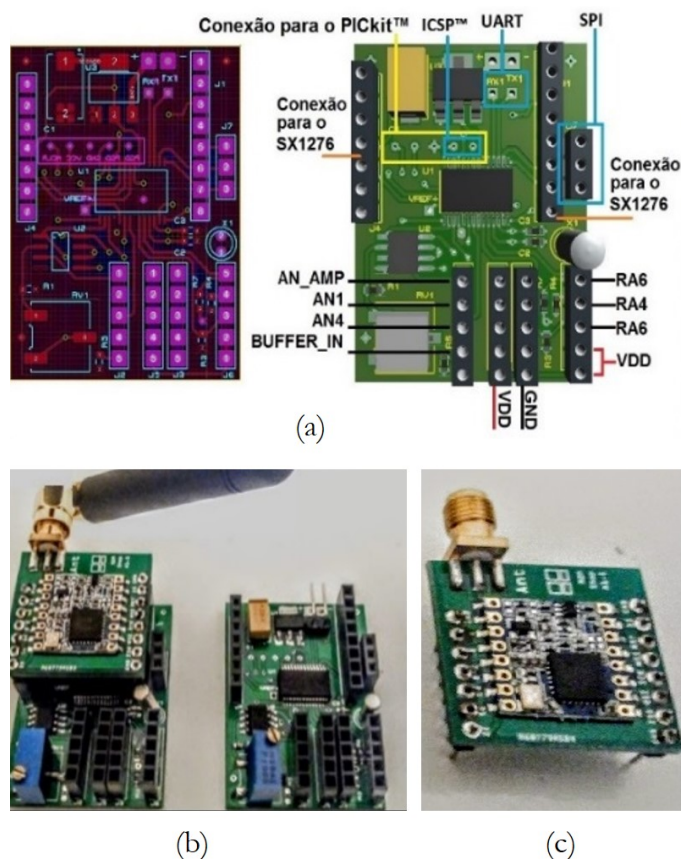


Figura 3. Confeção dos nós LoRaWAN (a) projeto PCB (b) montagem PCB + circuito montado (c) módulo SX1276.

O μ C PIC18LF26K22, contém 64 kB de memória interna, o suficiente para receber o protocolo LoRaWAN. Nos exemplos de aplicação da biblioteca LoRaWAN™ (Microchip, 2017) são utilizados somente μ C modelos PIC18LF46K22, porém é possível utilizar outros modelos listados ao final de (Microchip, 2016). Uma das contribuições deste trabalho é que, até o presente momento, o modelo PIC18LF46K22 é o único validado pela Microchip para ser utilizado com o LoRaWAN. Entretanto, o PIC18LF26K22 é utilizado e com resultados satisfatórios, atendendo aos objetivos propostos. Em (Microchip, 2017), tem-se os detalhes dos passos necessários para utilização do *plugin* LoRaWAN™ Library, que facilita o processo de configuração da biblioteca LoRaWAN nos μ Cs da Microchip.

O firmware do μC presente nos nós deve executar, sequencialmente, o envio da leitura dos sensores conectados ao nó, em intervalos de tempo pré-configurados, para os testes foi adotado um intervalo de 1 minuto, tempo suficiente para verificar alteração na leitura de sensores de resposta lenta como termorresistências. Após, abrem-se duas “janelas” de recepção. Encerrada a transmissão e possíveis recepções de dados, o nó deve entrar em modo *sleep*, estado em que o μC desliga o transceptor LoRa, seus periféricos e seu oscilador interno.

Na parte destinada ao envio de dados, dois valores distintos devem ser enviados em um *payload* fixo de 4 bytes, sendo os dois primeiros bytes utilizados para enviar a identificação da mensagem atual, obtido por meio da função `LORAWAN_GetUplinkCounter()`. Os dois bytes restantes são utilizados para enviar o valor de leitura do sensor ou qualquer outra informação de interesse. Para colocar o μC em modo *sleep* deve se utilizar a função `SLEEP()`, disponível no ambiente de programação do MPLAB® X IDE v5.15, utilizado para a programação do firmware do μC .

3.3 Execução do Servidor OPC UA

A maior contribuição deste trabalho consiste na execução de um servidor OPC UA embarcado em um *gateway* LoRaWAN, utilizando nativamente a distribuição Linux OpenWRT. Para alcançar esse objetivo é necessário o uso de métodos de *cross compile*, visando obter um programa escrito em linguagem C compatível com a arquitetura do processador utilizado no LPS8, o Atheros AR933. Por meio do SDK (*Software Development Kit*) para ambientes Linux, disponibilizado pelo projeto mantenedor do OpenWRT (Bursi, 2020). Esse SDK automatiza todo o processo de compilação e criação de pacotes, de forma ágil. Portanto, após a compilação e instalação do pacote, o servidor OPC UA é criado como um software do Linux OpenWRT OS e teste ocorre pela execução do comando `lorawan_opc_ua`, i.e. o nome da aplicação, conforme mostra a Figura 4.

```

Adding node list on the server: Total of 9 nodes read
[2020-06-14 17:39:28.938 (UTC-0300)] info/network TCP network layer listening on opc.tcp:
Making the temporary node list: Total of 9 nodes read

ID and AES key of each node
F1 7D BE 26 ---- 2A E6 DF F3 45 CF 24 F5 CE D8 FE E7 D6 3E 50 B1
1A BD BF 26 ---- 2A E6 DF F3 45 CF 24 F5 CE D8 FE E7 D6 3E 50 B1
2A BD BF 26 ---- 2A E6 DF F3 45 CF 24 F5 CE D8 FE E7 D6 3E 50 B1
    
```

Figura 4. Teste do *gateway* LoRaWAN/OPC UA executando no terminal.

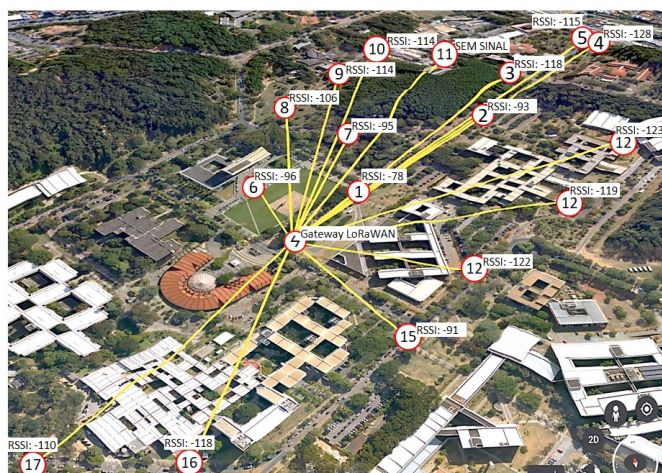
A Figura 4 apresenta parte da saída da execução do programa do servidor OPC UA funcionando. Nas linhas é possível observar o estabelecimento da seção e a adição dos nós ao servidor. Ao final, tem-se um resumo dos nós com os seus respectivos `AppKeys`. Tais mensagens exibidas no terminal são bastante úteis para depuração e verificação do funcionamento da aplicação. Uma vez confirmado o funcionamento da rede LoRaWAN através do *logfile* do *gateway*, é verificado a execução do servidor OPC UA através da conexão com clientes OPC UA.

4. RESULTADOS E DISCUSSÕES

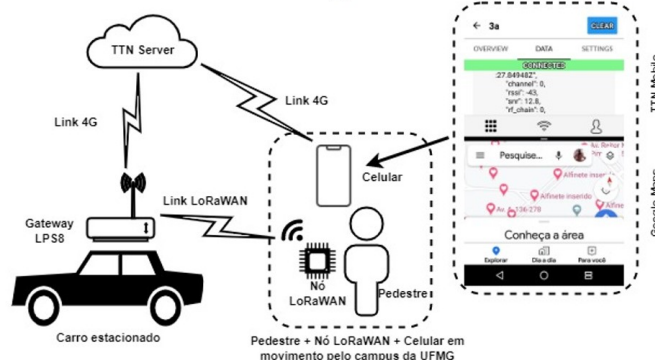
4.1 Teste de Alcance

Os testes de alcance foram realizados no campus Pampulha da Universidade Federal de Minas Gerais. A localização escolhida oferece uma variedade de áreas abertas e fechadas, bem como uma variedade de relevos que podem ser aproveitados para testar o alcance de transmissão em diferentes circunstâncias.

Para esse teste, o *gateway* se encontra em uma posição fixa, dentro de um carro munido de uma antena externa fixada ao teto. Dessa forma, movimentando-se pelo campus da UFMG, foram realizadas diversas medições do sinal, bem como o registro da posição geográfica do nó. Na Figura 5a é possível visualizar 17 pontos de medição geograficamente espalhados pelo campus da UFMG, onde todas as medições ocorreram em sequência, de 1 a 17, com o seu respectivo RSSI (*Received signal strength indication*) registrado em dBm, conforme mostra o mapa da Figura 5a.



(a)



(b)

Figura 5. Teste de alcance realizado no campus da UFMG (a) pontos de medição (b) esquemático do teste.

As medições realizadas com linha de visada direta ao *gateway*, em solo plano e sem obstáculos, apresentaram intensidades de até -115 dBm para uma distância de 781 m. O sinal mais baixo ocorre a 560m do *gateway*, essa medição apresenta a desvantagem de estar atrás de obstáculos como árvores e relevo irregular. E, ainda, há uma edificação de concreto relativamente próxima,

portanto, nesse ponto a intensidade foi de -130 dBm. A faixa de medição utilizada é de -139 a 0 dBm onde, abaixo de -139 dBm, tem-se o registro de "SEM SINAL", como é o caso do ponto de medição 11, mostrado na Figura 5a. A Tabela 1 apresenta alguns dos pontos obtidos no Teste de Alcance, que merecem destaque devido as diferentes condições do terreno, distância e intensidade do sinal.

Measured Point	Signal Power (dBm)	Distance (m)	Test conditions
1	-78	125	Campo aberto, terreno plano
5	-115	781	Campo aberto, terreno plano
10	-130	560	Vegetação densa entre gateway e nó, terreno irregular
12	-123	463	Vários prédios de concreto entre gateway e nó, terreno irregular

Tabela 1. Comparação entre potência do sinal, distância e condições do testes.

Na Figura 5b é mostrado um esquemático de como o teste de alcance foi realizado, *i.e.*, dentro de um carro estacionado no campus da UFMG se encontra o *gateway* LPS8 conectado à internet com o intuito de enviar todas as mensagens coletadas para a nuvem, registrando-as no famigerado servidor TTN (*The Things Network*). Assim, uma pessoa portando um nó LoRa e um *smartphone*, também conectado à internet, se movimenta pelo mesmo campus observando os dois aplicativos em execução (em divisão de tela no *smartphone*), o aplicativo TTN Mobile e Google Maps®. O nó LoRa se comunica via rede LoRaWAN com o LPS8, que por sua vez coleta uma mensagem enviada pelo nó e a transmite para o servidor TTN, via internet. O aplicativo TTN Mobile, no *smartphone*, acessa o servidor TTN, também via internet, possibilitando o monitoramento das mensagens recebidas do nó LoRa. A Figura 6 mostra os detalhes de instalações do *gateway* no carro utilizado para o teste de alcance:



Figura 6. Instalações do *gateway* no carro estacionado.

Pela Figura 6, pode-se observar que o *gateway* LPS8 se encontra dentro do carro conectado a um modem 4G USB para acesso à internet e, portanto, enviar os dados para o servidor TTN. Entretanto, uma antena externa é utilizada e afixada no teto do veículo, para melhoria da recepção do sinal proveniente do nó LoRa utilizado no teste. O LPS8 é energizado utilizando a bateria do próprio carro (12V_{CC}).

4.2 Conexão a um sistema SCADA via OPC UA

Verificada a eficiência da rede LoRaWAN, o teste de comunicação em uma rede OPC UA é realizado conectando-se

dois clients OPC UA listados anteriormente na mesma rede do servidor desenvolvido e executando no *gateway* LPS8. A Figura 7, mostra o servidor OPC UA rodando no momento em que dados são recebidos de dois nós diferentes.

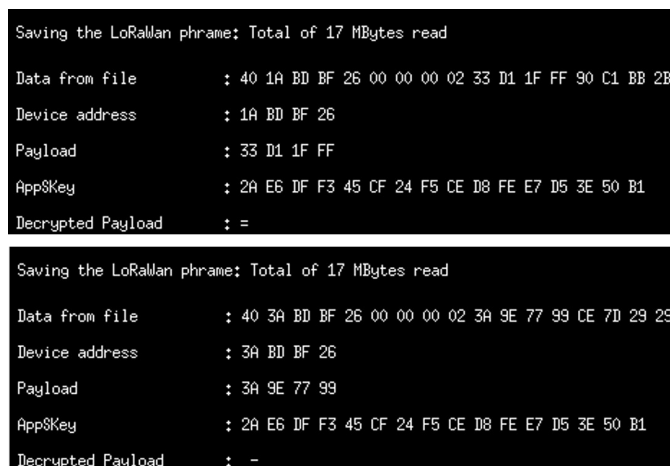


Figura 7. *gateway* LoRaWAN/OPC UA executando e recebendo dados.

A Figura 7 é gerada por meio de uma sessão de terminal SSH. Detalhes do processo de leitura do arquivo compartilhado e a identificação do que cada nó envia ao *gateway* são apresentados, mostrando a capacidade de conectividade e integração do *gateway* OPC UA desenvolvido, lendo e descriptografando o *payload* de múltiplos nós. De cima para baixo, é possível constatar o recebimento em sequência dos dados de diferentes nós por meio do campo *Device address*, enquanto o campo *Data from file* expõe os dados brutos lidos do arquivo compartilhado, onde é possível extrair o *payload* por meio de uma operação de descriptografia utilizando a chave *App\$Key*. Por último, ao observar a Figura 7 atentamente, é possível perceber os bytes presentes no arquivo compartilhado e com isso identificar todos os campos referente à camada de MAC.

Na Figura 8a, tem-se uma interface SCADA (*Supervisory, Control And Data Acquisition*) desenvolvida com o objetivo de demonstrar a viabilidade da integração entre nós sensores LoRa, via rede LoRaWAN, com o nível de operação industrial, *i.e.*, utilizando um sistema SCADA, nível de operação, comunicando-se via OPC diretamente com o nível de instrumentação. É importante salientar que esta configuração permite aumentar a adição de novos pontos de E/S sem que haja a intervenção em uma arquitetura de sistema de automação existente. Em outras palavras, a solução apresentada permite que haja expansão dos pontos de medições, de baixa criticidade para o processo (porém úteis a diversas análises) com um mínimo de intervenções.

A Figura 8a apresenta 10h de medição da temperatura de uma porção de água pura (Figura 8c), coletada pelo nó 26BFBD1A. O gráfico da Figura 8a é atualizado a cada novo recebimento de dados vindos do nó e nele é possível identificar vários momentos durante a medição da temperatura, em um volume de 500 mL de água. Pela curva da Figura 8a, pode-se observar que, inicialmente, a água está congelada e é aquecida até alcançar a temperatura ambiente de 29 °C. Neste ponto, o sistema é

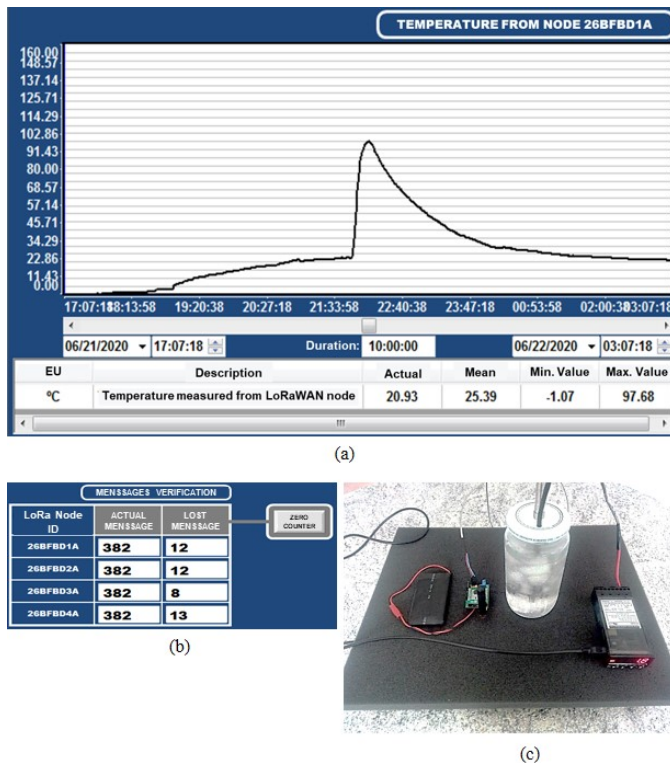


Figura 8. (a) interface SCADA com a medição de temperatura (b) contador de mensagens perdidas (c) pote de água com o sensor de temperatura conectado ao nó LoRa 26BFBD1A.

colocado sob aquecimento forçado até alcançar cerca de 97^o C, resfriando-se naturalmente.

Com intuito de demonstrar a eficiência da solução implementada, a Figura 8b apresenta um registro de pacotes perdidos na rede OPC UA, onde 4 nós LoRa (26BFBD1A a 26BFBD4A) estão conectados e trocando mensagens com o sistema SCADA. A sequência de mensagens recebidas por meio dos dois primeiros bytes do *payload* LoRaWAN possibilita identificar a quantidade de mensagens perdidas de cada nó, mediante a comparação do número da mensagem atual com o número da última mensagem recebida.

É possível perceber que o pior índice de perda foi de 3,4% e o melhor, 2,1%. Vale ressaltar que essas mensagens podem se perder por dois motivos: perda do pacote durante a transmissão ou problemas no software de integração. Assume-se aqui que as perdas ocorram devido a falhas na execução do software de integração LoRaWAN e OPC UA, embarcado no *gateway* LPS8. Um problema que ainda se encontra sob investigação.

5. CONCLUSÕES

Este trabalho procurou abordar todas as etapas necessárias para demonstrar a viabilidade da integração dos protocolos LoRaWAN e OPC UA de forma embarcada em um dispositivo de rede para, então, ser aproveitado como mais uma opção de comunicação para ambientes industriais, dentro do contexto da Indústria 4.0 e tecnologias IIoT. Os resultados obtidos nos testes da rede LoRaWAN demonstram as características promissoras do

protocolo, no que se refere ao alcance, custo e facilidade de implementação.

Os resultados obtidos no teste de alcance mostraram que a tecnologia pode ser utilizada em grandes pátios industriais onde a estrutura diversificada de tanques, tubulações e prédios de concreto não representam grandes impeditivos à coleta do sinal. E, por fim, os resultados obtidos no teste de conexão com o sistema SCADA demonstrou uma de suas principais características do OPC UA, *i.e.*, a independência de plataformas, por meio da compilação e execução de um servidor OPC UA embarcado no próprio hardware do *gateway* LoRAWAN, comunicando-se com o nível de operação industrial e contribuindo com a diminuição de custos em infraestrutura.

REFERÊNCIAS

- Bursi, A. (2020). OpenWrt. Using the SDK. https://openwrt.org/docs/guide-developer/using_the_sdk.
- Commserver (2010). UA System concepts. <https://cas.eu/index.php>.
- Compology (2017). A case study: Peninsula Sanitary Service Inc. (PSSI) implements container monitoring sensors. *Compology Waste Tech*.
- Dragino (2017). LPS8 LoRaWAN gateway user manual 1.0.2. *Dragino Technology*.
- LoRa Alliance (2015). LoRaWAN: What is it? A technical overview of LoRa® and LoRaWAN™.
- LoRa Alliance (2017). LoRaWAN™ backend interfaces 1.0 specification.
- LoRa Alliance (2018). LoRa Alliance Technical Committee Regional Parameters Workgroup. *LoRaWAN 1.0.3 regional parameters*.
- Mahnke, W., Leitner, H.S., and Damm, M. (2009). *OPC unified architecture*. Berlin. Springer, Berlin.
- Microchip (2016). Release notes for MPLAB® code configurator library LoRaWAN 01.10.00_beta.
- Microchip (2017). LoRaWAN™ library plug-in for MPLAB® code configurator user's guide.
- OPC Foundation (2017). OPC unified architecture: interoperability for industrie 4.0 and the internet of things.
- Open62541 (2016). Open62541 is an open source C (C99) implementation of OPC UA licensed under the mozilla public License v2.0. <https://open62541.org/>.
- Pursche, F. (2016). UA System concepts. <https://blog.bosch-si.com/industry40/container-4-0-smart-transport-high-seas/>.
- Rangel, R. (2014). Internet das coisas, nova revolução da conectividade. *Revista Inovação em Pauta, Rio de Janeiro*, (18), 4-7.
- Semtech Corporation (2015). AN1200.22: LoRa modulation basics - Applaction Note.
- Talla, V., Hessar, M., Kellogg, B., Najafi, A., and Smith, J.R. (2017). Lora backscatter: enabling the vision of ubiquitous connectivity. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, volume 1. New York.
- Unified Automation (2020). Address Space Concepts. Embedded OPC UA stack documentation: Unified Automation.
- Wich, J.P. (2020). OpenWrt. Welcome to the OpenWrt Project. <https://openwrt.org/>. Acesso em: 12 de Abril de 2020.