



Universidade Federal do Rio Grande do Norte – UFRN  
Centro de Ensino Superior do Seridó – CERES  
Departamento de Computação e Tecnologia – DCT  
Bacharelado em Sistemas de Informação – BSI

# **Desenvolvimento de uma plataforma de interconexão de código aberto para a Internet das Coisas**

**Dênis Robson Santos Pinheiro**

Orientador: Prof. Rivaldo da Silva Júnior

**Trabalho de Conclusão de Curso** apresentado ao Curso de Bacharelado em Sistemas de Informação como parte dos requisitos para obtenção do título de Bacharel em Sistemas de Informação.

Caicó, RN, 13 de dezembro de 2015

Catálogo da Publicação na Fonte  
Universidade Federal do Rio Grande do Norte - UFRN  
Sistema de Bibliotecas - SISBI

PINHEIRO, Dênis Robson Santos.

Desenvolvimento de uma plataforma de interconexão de código aberto para a Internet das Coisas. / Dênis Robson Santos Pinheiro. – Caicó, 2015.

39 f.: il.

Orientador: Rivaldo da Silva Júnior.

Monografia (Bacharel em Sistemas de Informação) – Universidade Federal do Rio Grande do Norte. Centro de Ensino Superior do Seridó - Campus Caicó.

1. Internet das Coisas. 2. Protocolo. 3. Redes de sensores sem fio. I. SILVA JÚNIOR, Rivaldo da. II. Título.

# Resumo

O conceito de Internet das Coisas, do inglês *Internet of Things* (IoT), é tido como o futuro da Internet, saindo da típica visão de comunicação para um paradigma mais amplo, onde dispositivos inteligentes conversarão entre si, independente da intervenção humana. Nesse novo paradigma não são apenas coisas conectadas à Internet, é um sistema inteligente que será capaz de oferecer, no seu ápice, funcionalidades que irão nos proporcionar uma melhor qualidade de vida. Entretanto, existem desafios para a IoT, como a fragmentação e a interoperabilidade. As Redes de Sensores sem Fio (RSSF), tecnologia base para este novo paradigma, são compostas geralmente por dispositivos de baixo poder de processamento e armazenamento. Isso faz com que as redes de sensores sem fio permaneçam isoladas da Internet. Com o crescimento da utilização do protocolo de Internet versão 6 (IPv6) nas redes pessoais sem fio de baixo poder de processamento, surge uma alternativa de interoperabilidade entre as RSSF e a Internet. Porém, a utilização dessa alternativa é invisível para uma rede externa e exige uma tradução de protocolo adequada para a comunicação entre as duas redes. Neste trabalho, é apresentado o desenvolvimento de uma plataforma aberta, com a utilização de ferramentas e tecnologias livres que torna possível uma comunicação direta entre Redes de Sensores sem Fio e os dispositivos da Internet.

**Palavras-chave:** Internet das Coisas; Redes de Sensores sem Fio; Protocolo.



# Abstract

The concept of Internet of Things is seen as the future of the Internet, changing from the typical vision of communication to a broader paradigm, where smart devices will talk to each other, independent of human intervention. In this new paradigm not only things are connected to the Internet, it is an intelligent system that will be able to offer, at its summit, features that will provide us a better quality of life. Although, there are challenges for the IoT such as, as fragmentation and interoperability. The Wireless Sensor Networks (WSN), base technology for this new paradigm, are usually composed by low processing power and storage devices. This causes the wireless sensor network to be isolated from the Internet. With the growth of the use of Internet Protocol version 6 (IPv6) in low-power wireless personal area networks, comes an alternative interoperability between WSN and the Internet. However, the use of this alternative is invisible to an external network and requires a suitable protocol translation for communication between the two networks. In this work, show the development of an open platform, with the use of free tools and technologies that enables direct communication between sensor networks and Internet devices.

**Keywords:** Internet of Things; Wireless Sensor Network; Protocol.



# Sumário

<b>Lista de Figuras</b>	<b>iii</b>
<b>Lista de Tabelas</b>	<b>v</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Objetivos . . . . .	4
1.2 Organização do Documento . . . . .	4
<b>2 Revisão Bibliográfica</b>	<b>7</b>
2.1 Internet das Coisas . . . . .	7
2.1.1 A evolução da IoT . . . . .	7
2.1.2 Desafios . . . . .	9
2.1.3 Proposta de Arquitetura de Rede para IoT . . . . .	10
2.2 <i>Internet Protocol</i> - IP . . . . .	11
2.2.1 IPv4 . . . . .	11
2.2.2 IPv6 . . . . .	11
2.3 Redes de Sensores Sem Fio . . . . .	14
2.3.1 Padrão 802.15.4 . . . . .	14
2.3.2 6LoWPAN . . . . .	16
2.3.3 Trabalhos relacionados . . . . .	17
<b>3 Desenvolvimento</b>	<b>19</b>
3.1 Procedimentos Metodológicos . . . . .	19
3.1.1 Caracterização da pesquisa . . . . .	20
3.1.2 Estudo técnico . . . . .	21
3.1.3 Desenvolvimento da plataforma . . . . .	21
3.2 Visão geral . . . . .	21
3.3 Planejamento . . . . .	21
3.3.1 Hardware . . . . .	22
3.3.2 Software . . . . .	23
3.4 Gateway . . . . .	24
3.4.1 Hardware . . . . .	25
3.4.2 Software . . . . .	27
3.5 Resultados . . . . .	30
3.6 Considerações Finais . . . . .	31
<b>4 Conclusões</b>	<b>33</b>
4.1 Trabalhos futuros . . . . .	33





# Lista de Figuras

1.1	Sistemas interconectados formando a IoT. . . . .	2
2.1	Explosão do número de possíveis dispositivos conectados à Internet. . . . .	8
2.2	Interação entre nó IPv6 e nó da RSSF através do <i>gateway</i> . . . . .	11
2.3	Cabeçalhos do IP na versão 6 e 4 e suas alterações. . . . .	12
2.4	Processo de autoconfiguração de nó utilizando o EUI-64. . . . .	13
2.5	Exemplos de topologias para redes 802.15.4. . . . .	15
2.6	Tradução do protocolo a nível de enlace. . . . .	16
3.1	Arquitetura de <i>Gateway</i> . . . . .	24
3.2	Captura de tela da interface do Arduino IDE. . . . .	26
3.3	Desenho gráfico do <i>gateway</i> e sensor/atuador. . . . .	27
3.4	Captura de tela do XCTU para configuração do Xbee roteador. . . . .	28
3.5	Captura de tela do XCTU para configuração do Xbee coordenador. . . . .	28
3.6	Fluxo de mensagem entre nó IP, <i>gateway</i> e objeto inteligente. . . . .	30
3.7	Imagem real do <i>gateway</i> ao lado do sensor/atuador. . . . .	31



# Lista de Tabelas

3.1	Pinagem de módulo Ethernet HanRun . . . . .	25
3.2	Pinagem de módulo comunicador XBee . . . . .	27



# Capítulo 1

## Introdução

*“The most profound technologies are those that disappear”*

*Mark Weiser*

Através das extensas vantagens proporcionadas pela tecnologia, a sociedade está caminhando para um paradigma onde tudo está conectado. A Internet está bem difundida, tanto de modo cabeada como sem fio, além disso, existem grandes instituições que são responsáveis por organizar e propiciar uma melhoria no funcionamento da Internet. Tem-se por exemplo a IETF (Internet Engineering Task Force), que influencia, através da produção de documentos técnicos de alta qualidade, nos profissionais que planejam e gerenciam a Internet. O avanço na capacidade de processamento dos dispositivos computacionais, a redução de custos desses materiais e a comunicação sem fio têm estimulado o desenvolvimento e uso de sistemas inteligentes em áreas associadas com o futuro da Internet.

No início dos anos 90, o norte-americano [Weiser \(1991\)](#) previa, já naquela época, um aumento na disponibilidade de serviços e funcionalidades da computação para os usuários finais, porém a visibilidade destes serviços seria a menor possível. Para ele um computador com grande poder de processamento prende a atenção do usuário em uma simples máquina, portanto, esses dispositivos devem ser dispostos de forma distribuída em um ambiente e conectados entre si, a ideia dos computadores pessoais em si era equivocada. Foi nesse contexto que ele declarou a frase “As mais profundas tecnologias são aquelas que desaparecerão” ([Weiser, 1991](#)). Os computadores deveriam integrar-se de forma natural ao nosso ambiente, seu funcionamento deve ser imperceptível, tornando mais fácil as tarefas do dia a dia. Isto é, o computador convencional como o conhecemos será substituído por diversos dispositivos computacionais completamente integrados ao mundo a nossa volta. Mais tarde, essas ideias seriam compreendidas como o início de um novo paradigma computacional, a Computação Ubíqua, do inglês, *Ubiquitous Computing*.

A Computação Ubíqua, nesse contexto, não significa apenas computadores que podem ser levados para outros lugares, por exemplo aeroportos e praças. Através da mudança na forma de interação homem/máquina e da imersão da capacidade computacional em atividades do cotidiano das pessoas, os usuários interagirão agora com os *Smart Objects* (Objetos Inteligentes), computadores embarcados em aparelhos de uso comum. Por exemplo, os carros estarão conectados à Internet, como já é realidade em alguns veículos recentemente lançados. Assim, o ambiente terá vários computadores embarcados em seus objetos, permitindo a interação do usuário com estes equipamentos de forma inteligente,

sem necessitar de intervenção de forma convencional por parte do usuário. Consequentemente, esses ambientes compostos por objetos inteligentes também é chamado de ambiente inteligente ([Vasseur and Dunkels, 2010](#)).

Um dos novos conceitos associado com o futuro da Internet é o que chama-se de Internet das Coisas, do inglês *Internet of Things* (IoT). A Internet das Coisas é um conceito tecnológico que sai do típico paradigma da comunicação atual para uma visão bem mais ampla, onde não apenas as pessoas estão conectadas, mas também “coisas” conversarão umas com as outras ([Coetzee and Eksteen, 2011](#)). Essa mudança terá como resultado a integração dos Sistemas Embarcado e Sistemas de Informação, assim, terá uma grande oferta de meios para manter-se conectados com o ambiente, podendo acessar serviços, interagir com as pessoas e dispositivos.



**Figura 1.1:** Sistemas interconectados formando a IoT.

Fonte: [Khan et al. \(2009\)](#)

Em [Weiser \(1991\)](#) vários filósofos, economistas e cientistas da computação afirmam que, em essência, apenas quando os objetos tornam menos perceptíveis no cotidiano, ficamos livres, deixamos de pensar no uso deste objeto e pensamos mais em nossa meta principal. Por exemplo, ao olharmos para um semáforo de trânsito, absorvemos as informações transmitidas por ele sem consciência de realizar o ato ou leitura do sinal, preocupando apenas em chegar no local de destino. Então, com a tendência da imersão da tecnologia no cotidiano da sociedade e do desaparecimento dela sendo consequência fundamental da psicologia humana, existe uma quantidade inimaginável de oportunidades de negócios envolvidas nos estudo e utilização de aplicações que usem novos paradigmas da Internet.

Uma das bases de pesquisa que contribuem para o desenvolvimento dos ambientes inteligentes é o estudo das Redes de Sensores Sem Fio (RSSF). Uma RSSF é uma rede composta por dispositivos computacionalmente limitados e que tem como principal objetivo realizar a coleta de dados de um ambiente físico. Outro objetivo que pode ser citado é capacidade desses sensores também serem atuadores, controlando o ambiente por meio de

alteração do estado dos dispositivos. Isso se dá através de sensores dos mais diversos tipos distribuídos em um região de interesse, conectados de modo sem fio. Esses dispositivos são tipicamente alimentados por baterias, tendo seu tempo de operação podendo variar de semanas ou até a anos de forma autônoma (Loureiro, 2003). Dessa forma, a evolução e resolução de problemas existentes nas RSSF podem ser utilizadas na IoT.

Porém, enquanto as RSSFs são utilizadas principalmente com o objetivo de monitorar variáveis que caracterizam um ambiente, a IoT tem um conceito bem mais amplo, não quer apenas conectar coisas, mas sim integrar esses objetos à Internet. A capacidade de tornar cada dispositivo individualmente endereçável na web Internet favorece o compartilhamento de informações entre sistemas. Sendo assim, essas coisas devem cooperar com os *Web Services* (WS) já em funcionamento (Gomes, 2010).

Os objetos inteligentes são compostos por um aparelho comum (ex. geladeira) e por um poder de processamento agregado a ele, sendo desejável a necessidade que ele tenha um tamanho reduzido para que o dispositivo computacional não interfira no funcionamento normal do utensílio. Como o poder computacional do *hardware* é limitado, tendo um processamento baixo e pouco espaço de armazenamento (Borges Neto et al., 2010), esses aparelhos não possibilitam a implementação da pilha de protocolos e aplicações como é desenvolvido nos computadores pessoais comumente utilizados na Internet. Portanto, isto caracteriza uma situação onde é necessário uma nova estrutura de rede desses objetos inteligentes, no que seria um fragmento da Internet. Porém, Segundo Diogo et al. (2014), a fragmentação é algo que não pode acontecer. Pois, a Internet deve ser uniforme, os dados e informações devem ser compartilhados com todos, usando uma arquitetura onde a complexidade computacional seja implementada nas bordas da rede. Os elementos intermediários da comunicação devem realizar apenas tarefas simples, como encaminhamento de dados.

Quanto a esses problemas e a exigência de ter uma Internet sólida, pode-se fazer um paralelo com um problema bem comum existente nas redes IP (*Internet Protocol*), que é a utilização do NAT (*Network Address Translation*). O NAT é uma técnica que isola as redes privadas, “reservando algumas faixas de endereços dentro de cada uma das classes padrões para uso apenas local, sem comunicação com a Internet” (Brito, 2013). Ou seja, uma solução temporária para o esgotamento dos endereços IP, em sua versão 4, que trouxe problemas no processo de comunicação fim-a-fim, ferindo os princípios arquiteturais da Internet.

Com a vinda do IPv6 (*Internet Protocol, version 6*), não existe mais esse problema da falta de endereços, conseqüentemente não é necessário utilizar o NAT, pois todos os *hosts* de uma rede têm seus endereços públicos disponíveis. Isso implica no melhor funcionamento de diversas aplicações na Internet.

O surgimento do IPv6 prova que a comunicação IP pode ser utilizada para diversos fins e em uma diversidade de dispositivos com características diferentes. Nesse contexto, sua utilização traz uma série de vantagens: (i) ferramentas já existentes para configuração, gerenciamento e tratamento de problemas nas redes IP podem ser utilizadas; (ii) Outros protocolos que funcionam sobre IP, TCP ou UDP podem ser aproveitados (Moreiras, 2013); Para este trabalho, e (iii) oferece uma alternativa para as RSSF que podem comunicar-se com os sistemas e ferramentas de IP tradicionais. Essa alternativa é a compressão do cabeçalho IPv6 em pacotes pequenos que carregam apenas informações fundamentais. Ela permite que o cabeçalho do IPv6 fique mais curto removendo informações redundantes e desnecessárias. No entanto, a utilização desses endereços curtos dentro da rede de

sensores, não é visível para o mundo exterior e, por conseguinte, uma tradução adequada é necessária. Para isso, é fundamental desenvolver uma arquitetura de *gateway* que conecte o mundo das RSSFs à Internet, permitindo acessibilidade direta à primeira.

## 1.1 Objetivos

Tem-se por objetivo geral, com essa pesquisa, o desenvolvimento de uma plataforma que torne possível uma comunicação direta entre Redes de Sensores e os dispositivos da Internet utilizando IPv6. Devido à competição entre grandes empresas na busca de ser precursora na documentação de padrões de gerenciamento da IoT, existe a possibilidade de que cada fabricante proponha diferentes maneiras de realizar esta tarefa, criando padrões fechados e implementações incompatíveis com o sistema de protocolos utilizados na Internet. Porém, esta filosofia vai diretamente de encontro aos princípios da Internet, que é composta por padrões livres e abertos, para que tanto organizações como indivíduos possam acessá-los. Para [Vasseur and Dunkels \(2010\)](#), a adoção e popularização dos objetos inteligentes têm sido dificultada pela grande quantidade de sistemas fechados e proprietários, dificultando até mesmo a integração de soluções. Com isso, tem-se vantagens por exemplo em, (i) respostas rápidas em possíveis falhas de implementação, (i) qualquer indivíduo pode contribuir para o projeto, e (iii) será uma contribuição para o meio científico, onde todos terão acesso às informações do projeto. Portanto, este trabalho utilizará ferramentas e tecnologias livres para o seu desenvolvimento, tornando a adoção deste projeto mais ampla na comunidade da IoT.

A plataforma será formada por uma camada de *software*, que executará a função de tradução de mensagens entre os protocolos usados na comunicação, e outra camada de *hardware*, que fornecerá módulos de comunicação que permitirão a conexão, em baixo nível, entre os dispositivos da Internet e dos Sistemas Embarcados. Além disso, este trabalho terá como objetivos específicos:

1. Especificar protocolos que podem ser usados na comunicação das redes de sensores com os WS;
2. Implementar protocolo da camada de rede a fim de criar uma interface de comunicação com os objetos inteligentes;
3. Adaptar a pilha IPv6 no Sistema Operacional do *gateway*, criando uma interface de comunicação com a Internet, assim tornando acessível para comunicar com objetos inteligentes.

## 1.2 Organização do Documento

Além deste capítulo, este Trabalho de Conclusão de Curso está organizado em outros quatro capítulos da seguinte forma: o Capítulo 2 traz uma revisão bibliográfica no que envolve IoT, nele explica os conceitos relacionados à IoT, também relata algumas preocupações sobre a utilização desse novo paradigma de Internet e uma arquitetura de rede característica para a comunicação entre os objetos inteligentes e a Internet. Outro ponto abordado é a utilização do protocolo IP no contexto de IoT e suas contribuições. Por fim, retrata as RSSF, explicando as características desta rede, e expondo todas as características e padrões em que as RSSF está baseada.

O Capítulo 3.1 retrata os procedimentos metodológicos adotados para obter resultado satisfatório do trabalho. Serão apresentados detalhes de planejamento e análises que po-



dem afetar na conclusão da produção da plataforma de integração. Outros pontos abordados neste capítulo são os possíveis equipamentos utilizados no decorrer do desenvolvimento do projeto.

O Capítulo 3 irá explicar como foi desenvolvido o trabalho até que fosse alcançado os objetivos do trabalho, bem como os aparelhos utilizados como as implementações realizadas para a tradução de protocolos. Por fim, a última Seção deste Capítulo, o 4, descreve o cronograma do projeto. Nele estão organizados as atividades que serão realizadas de acordo com o tempo durante o desenvolvimento do projeto.



## Capítulo 2

# Revisão Bibliográfica

*“The Internet of Things has the potential to change the world,  
just as the Internet did. Maybe even more so.”*

*Kevin Ashton*

Este Capítulo está organizado da seguinte forma, na Seção 2.1 descreve-se porque as tecnologias conectadas à Internet estão passando para uma nova forma de comunicação e interação com o ambiente. Além disso, também são apresentados os desafios encontrados para que esta nova fase possa ser concretizada no meio tecnológico e uma arquitetura que caracteriza a IoT. Já na Seção 2.2, é explanado o protocolo de comunicação de rede IP. Nela está descrita toda a motivação da criação do novo protocolo da Internet e suas contribuições para a IoT. Por fim, na Seção 2.3 é abordado sobre o tema no qual muitos afirmam ser a base da Internet das Coisas, ela explicará os principais conceitos de padrões e normas a serem seguidos para comunicação de dispositivos com recursos computacionais limitados.

## 2.1 Internet das Coisas

### 2.1.1 A evolução da IoT

Para entender o surgimento da IoT é indispensável organizar a cronologia da Internet em três fases: (i) Internet das Máquinas; (ii) Internet das Pessoas e (iii) Internet das Coisas (Brito, 2013). Ela iniciou como a Internet das Máquinas, uma rede global com serviços como a World Wide Web que foi construída em cima de uma plataforma tradicional. Nesta época a Internet foi construída para conectar máquinas, estas sendo implantadas em prédios (onde originou o termo *sites*).

Ao longo dos anos a Internet foi evoluindo para a Internet das Pessoas, criando conceitos como a Web Social (Web 2.0). Nessa era as pessoas estão conectadas às redes sociais em qualquer lugar. Com a enorme disseminação dos dispositivos móveis a partir da virada do século (2000), o elemento mais importante deixa de ser a máquina e passa a ser as pessoas. Isto também ficou evidente com o crescimento exponencial dos serviços sociais. Pontos importantes tais como as 500 milhões de contas criada por pessoas no Facebook, também o Hotmail.com, que conta com mais de 300 milhões de *e-mail* ativos (Coetzee and Eksteen, 2011), representa de forma significativa essa etapa de Internet das Pessoas.

Com a expansão da Internet, a próxima fase é aquela em que qualquer coisa estará conectada à Internet. Dispositivos são cada vez mais equipados com sensores e atuadores.

Esta combinação cria um ambiente onde dispositivos têm capacidade de agir, detectar e computar, de forma a tornar o ambiente inteligente. Da mesma forma, objetos físicos são cada vez mais equipados com *tags* (marcadores) onde podem ser detectados por RFID (*Radio-Frequency IDentification*) ou examinados via *QR-Code* (*Quick Response*) por dispositivos móveis. Por exemplo, os televisores, as geladeiras, as cafeteiras, os fogões e os carros farão parte da Internet. Santucci (2010) afirma que de 50 a 100 bilhões de dispositivos serão conectados à Internet até 2020. Também que, neste novo paradigma, os objetos em rede serão tantos que irão transbordar a linha entre *bits* e átomos. Outro dado importante em relação à explosão de possibilidades de conexões é mostrado na Figura 2.1.

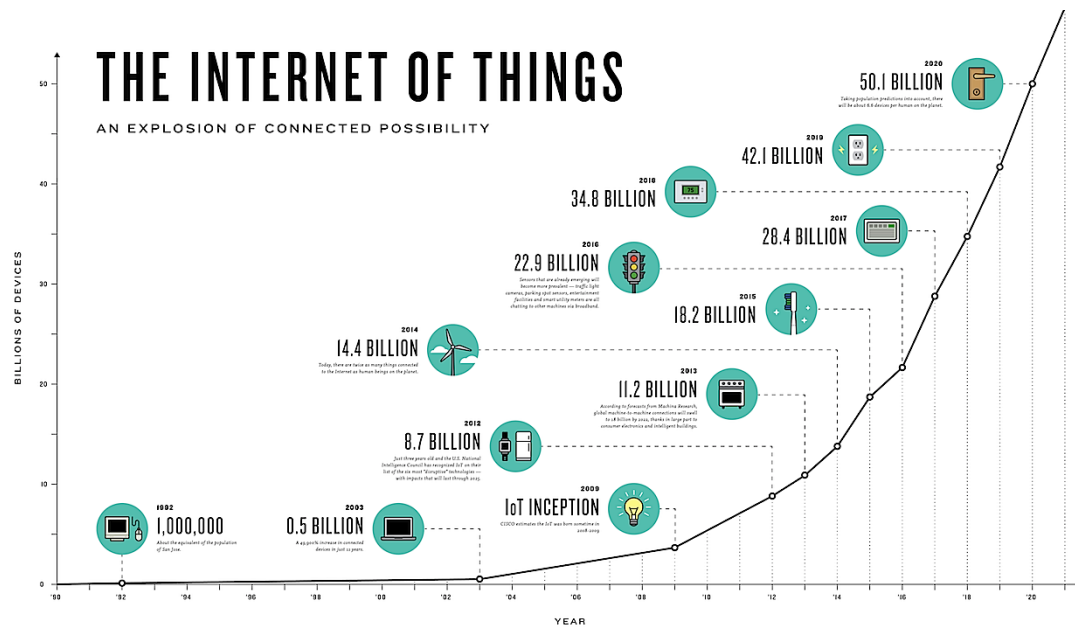


Figura 2.1: Explosão do número de possíveis dispositivos conectados à Internet.

Fonte: Harpham (2014)

Essa evolução é possível devido à evolução da tecnologia, sendo mais específico, aos estudos dos sistemas embarcados e das RSSF. Nesse sentido, esse envolvimento dos objetos na Internet através dos objetos inteligentes é o que chama-se do novo paradigma da Internet das Coisas.

O termo “*Internet of Things*” foi incluída no meio tecnológico pelo britânico Ashton (2009) e seus colegas de laboratório, em 1999. Ashton (2009) afirmou “Eu posso estar errado, mas eu tenho quase certeza como a frase ‘Internet of Things’ iniciou sua vida como um título de uma apresentação que eu fiz na *Procter & Gamble* (P&G) em 1999”. Ele é co-fundador de um laboratório no Instituto de Tecnologia de Massachusetts (MIT), o *Auto-ID Center*, e também, pioneiro na criação de um sistema de padrão global para RFID e outros sensores. O laboratório no qual ele fazia parte, tinha como meta a criação de uma rede de computadores conectados à objetos, não apenas *hardware*, mas na verdade tudo que é necessário para contribuir com este novo paradigma. É importante notar que a filosofia do laboratório é bem condizente com o que hoje é atribuído à IoT. Eles não desejam criar outra rede global, mas sim uma rede que contribua com a Internet, permitindo rastrear itens e compartilhar informações.

Todo o meio científico considera que os desafios da IoT são diversos e de alto nível. Portanto, existem algumas perspectivas em torno desse novo paradigma que podem ser mencionadas (2.1.2). A comunidade acadêmica examina os requisitos técnicos e o que pode ser desenvolvido para enfrentar os desafios da sociedade atual com a tecnologia da Internet das Coisas, por exemplo: sistemas de monitoramento de saúde para acompanhar o envelhecimento das pessoas (Diogo et al., 2014); sistemas distribuídos para ajudar a prevenção de desastres naturais e reagir de forma mais adequada; rastreamento e acompanhamento para ajudar a reduzir o congestionamento de tráfego (Vasseur and Dunkels, 2010); e na obtenção de informações da vida útil do produto para melhorar na reciclagem. Assim, pode-se perceber que todas estas aplicações estão relacionadas e contribuem para o que se chama de “Planeta Verde” (Coetzee and Eksteen, 2011).

### 2.1.2 Desafios

Para o avanço desse novo paradigma, são encontradas grandes barreiras no âmbito social. Em uma era onde todas as coisas estão ligadas, a comunicação de dados em um ambiente monitorado por sensores e atuadores, como também as pessoas que ali envolvidas diretamente, necessitam ter um tratamento de medidas de segurança privilegiado. O grande número de dados transmitidos à Internet, fornecendo todos os tipos de informações, é um aspecto que a IoT deve priorizar. Encontrar maneiras de armazenar de forma confiável e interpretar as massas de dados através de aplicações escaláveis continuam a ser um grande desafio tecnológico. O direito do indivíduo à privacidade deve ser protegido.

Na literatura, diversos autores demonstram um certo grau de preocupação quanto à maneira que os dados obtidos serão utilizados. Santucci (2010) afirma que na ITU (*International Telecommunication Union*), agência especializada das Nações Unidas para tecnologias de informação e comunicação, identifica os desafios mais importantes que precisam ser enfrentados para explorar plenamente o potencial da IoT: padronização e harmonização, privacidade e questões sócio-éticas. Há um equilíbrio a ser atingido através do uso dessa tecnologia: de um lado terá a proteção à privacidade, e de outro a comodidade obtida através do uso dela. Como visto na Figura 2.1, o aumento massivo na quantidade de dispositivos conectados à Internet trará uma crescente dependência de conectividade. Atualmente, esta dependência a equipamentos e computadores já é facilmente compreendida. Portanto, as grandes instituições e as autoridades públicas têm uma responsabilidade de assegurar que a IoT criará impacto positivo no crescimento econômico e resolverá os problemas sociais.

Atualmente, muitos fabricantes estão criando soluções verticais, uma parcela de contribuição para uma Internet fragmentada, usando suas próprias tecnologias e serviços inacessíveis. Padronização das tecnologias é importante, uma vez que irá levar a uma melhor interoperabilidade, diminuindo assim as barreiras de conexão entre as bordas. Coetzee and Eksteen (2011) afirma que os padrões e tecnologias devem ser criados para tornar essa “Intranet das Coisas” em uma “Internet das Coisas”.

Um outro grande desafio, como citado anteriormente, é lidar com a limitação do *hardware* nos dispositivos embarcados. Uma das principais motivações na utilização do IPv6 no contexto da IoT é a grande quantidade de IPs disponíveis, permitindo acesso direto entre os *hosts*, não ferindo um dos princípios fundamentais da Internet, a comunicação fim a fim. Porém, esse identificador de nós do IPv6 ainda é consideravelmente grande porque o dispositivo que armazenará tais pacotes possui pouca memória disponível para isto.

Além disso, enquanto a especificação do IPv6 requer um valor máximo de MTU (*Maximum Transmission Unit*) de 1280 *bytes* para as suas camadas superiores, para os dispositivos que atuarão nas RSSF, terão que operar com apenas 127 *bytes*, que é valor do MTU deste protocolo de enlace, podendo deixar, no pior caso, apenas 81 *bytes* para o IPv6. Destes 81 *bytes* após remover os 40 *bytes* do cabeçalho IPv6, restarão apenas 41 *bytes* e, em caso da utilização do TCP como protocolo de transporte, outros 20 *bytes* serão utilizados para este protocolo (Vasseur and Dunkels, 2010). Neste contexto, para a camada de aplicação restarão apenas 21 *bytes*.

Sendo assim, para que ambientes inteligentes, computação ubíqua e conectividade total entre objetos inteligentes por meio da Internet torne-se uma realidade, ainda faz-se necessário que muitos esforços sejam empreendidos, tanto por pesquisadores quanto empresas, com o intuito de encontrar soluções eficientes e otimizadas para as necessidades deste novo cenário.

### 2.1.3 Proposta de Arquitetura de Rede para IoT

A interação entre o usuário final com um nó de uma RSSF envolve três elementos principais, são eles: o nó sensor/atuador (objeto inteligente), o *gateway* e o *host* IPv6 (Rao.S et al., 2009). Caso a arquitetura de Rede tenha o objetivo de ser usada para uma aplicação com fins de telemetria ou qualquer outra aplicação que necessite de armazenamento dos dados, facultativamente, é necessário um quarto elemento. Este quarto elemento consiste em um servidor onde será realizado a função de armazenar todos os dados decorrentes da rede de sensores, contribuindo na melhoria das tomadas de decisões e eficiência para os Sistemas de Informação (Medeiros, 2006). De acordo com a Figura 2.2 pode-se verificar componentes que constituem a integração entre os Sistemas Embarcado com os Sistemas de Informação:

1. *Host* IPv6 - Este é o elemento que irá compor a borda da rede, ou seja, o ponto mais extremo. É uma máquina comum conectada à Internet que enviará ou receberá dados de forma direta aos/dos nós da RSSF;
2. *Gateway* - Ele é a porta de saída dos dados em relação aos nós da RSSF. Terá a responsabilidade de habilitar o encaminhamento dos pacotes entre RSSF e a rede externa, IPv6. Portanto, é essencial uma tradução entre as duas redes heterogêneas. Para isso, duas pilhas de protocolos devem ser configuradas no *gateway*;
3. Sensor/atuador sem Fio - São dispositivos pequenos capazes de coletar informações dos ambientes físicos. Neles é necessário criar uma camada de adaptação para a pilha IPV6, sendo assim, possível a comunicação entre os dois lados através de um único protocolo;
4. Servidor Web - Este é o elemento que representa os Sistemas de Informação. Sua utilização é opcional em uma arquitetura que caracteriza IoT, ele pode ser configurado para fazer parte do *gateway*. Porém, em algumas outras arquiteturas é uma entidade separada (Rao.S et al., 2009).

Tendo abordado muitos conceitos acerca de IoT, na próxima Seção são explanados conceitos, características e vantagens sobre a utilização do IP para o desenvolvimento deste projeto.

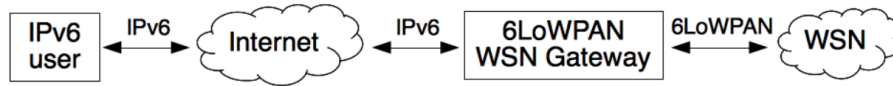


Figura 2.2: Interação entre nó IPv6 e nó da RSSF através do *gateway*.

Fonte: Campos et al. (2011)

## 2.2 *Internet Protocol - IP*

Para a Internet das Coisas é importante notar que é necessário o acesso direto aos objetos inteligentes. Portanto, uma característica importante para o sucesso do desenvolvimento desta plataforma de integração entre os Sistemas Embarcados e os Serviços Web é a interoperabilidade na comunicação entre os dispositivos. Com a utilização do protocolo IP, seguindo a arquitetura TCP/IP (Socolofsky and Kale, 1991), a Internet permanece uniforme. Como dito, os serviços das tecnologias de acesso serão resolvidos nas camadas física e enlace, consequentemente a integração que permitirá a comunicação entre usuários e objetos inteligentes será através da unicidade de apenas um protocolo, o IP.

### 2.2.1 IPv4

De acordo com Postel (1981), o endereço IP é um identificador de *host* a nível de camada de rede composto por 32 *bits*, esses separados em quatro blocos de 8 *bits*, também conhecido como octetos. Na época em que o protocolo foi desenvolvido, tinha como objetivo interligar instituições como governos, faculdades e bases militares, portanto era baseado em endereços hierárquicos onde há um prefixo para identificar uma rede, e o sufixo para determinar um *host* específico dentro da rede. Com os 32 *bits* de espaço para endereçamento, é possível identificar  $2^{32}$  nós, resultando em aproximadamente 4 bilhões e 300 milhões de endereços. Na época, parecia um número suficiente para todos os *hosts*, as pessoas não tinham computadores em casa, assim ninguém poderia imaginar que esses endereços um dia iriam esgotar-se.

### 2.2.2 IPv6

Com o surgimento do IPv6 (Deering and Hinden, 1998), tornou-se viável o desenvolvimento de sistemas e serviços para a Internet das Coisas. Isto seria impossível com a indisponibilidade de endereço IP no mundo. Segundo a IANA (*Internet Assigned Numbers Authority*), órgão responsável pela distribuição de alguns blocos às autoridades regionais, em 2011 o estoque de endereços acabaram. O IPv6 é um protocolo que possui 128 *bits* no campo de endereço, assim em relação ao IPv4, é aproximadamente 79 trilhões de trilhões de vezes maior. Ter uma noção da imensidade da quantidade de endereços disponíveis no IPv6 é uma atividade difícil, por isso é importante ser delicado em algumas formas de explicar o IPv6.

Um equívoco que não pode existir é pensar que, pelo fato do IPv6 ter 4 vezes mais *bits* que o IPv4, não significa que a capacidade que ele tem de endereçar *hosts* é de apenas 4 vezes mais, afinal, a cada bit adicionado na lista de endereços, duplica o número de combinações possíveis. Assim, pode-se entender também que, a menor rede local (/64) de endereçamento IP na versão 6 tem uma capacidade ainda maior de identificar cada *host* do que os 4,3 bilhões do IPv4 (/32). Ainda, para tentar explicar mais o real tamanho

de endereçamento da nova versão de protocolo da Internet, Brito (2013) apresenta uma forma diferente de mensurar isso. Recentemente, foi constatado que o mundo passou das 7 bilhões de pessoas, mas imaginando que o mundo tenha 10 bilhões de habitantes, com esses dados, pode-se dizer que cada habitante da Terra poderia ter  $3,5 \times 10^{27}$  endereços para fins individuais.

Além de todos estes números, esta nova definição do protocolo IPv6 pode ser caracterizada pelo retorno aos princípios fundamentais da Internet (Vasseur and Dunkels, 2010). Alguns destes princípios são: (i) a otimização dos cabeçalhos IPv6, com a disposição dos campos presentes nos cabeçalhos da camada de rede e a fixação do tamanho do cabeçalho base (40 bytes), são modificações que permitirão um melhor desempenho durante o tratamento dos pacotes IPv6, tarefa esta essencial para o tipo de rede que está descrita neste trabalho; (ii) segurança embutida com IPsec (*IP Security Protocol*) (Frankel et al., 2011), uma maior importância com a segurança e autenticação durante a comunicação, preocupações ainda maiores quando os dados estão vinculados com a privacidade de seus usuários, como suas preferências e características pessoais (Jayanthi and Rabara, 2010); e (iii) a autoconfiguração dos nós, que é uma característica fundamental para a utilização em grande escala dos objetos inteligentes, bem como na impossibilidade de manter um elemento encarregado de configurar os objetos em um ambiente inteligente, que poderão constituir-se de muitos componentes, até milhares, tornando esta tarefa muito complexa de ser alcançada (Schrickte, 2013).

Em relação ao IP na sua versão 4, o IPv6 alterou alguns campos do cabeçalho, porém outros foram mantidos ou renomeados. O principal benefício na otimização dos cabeçalhos IPv6 foi a fixação do tamanho do cabeçalho, porém outros ajustes complementaram esta melhoria. Conforme verificado na Figura 2.3, o campo *Internet Header Length* do IPv4 foi retirado, isto devido o cabeçalho do IPv6 ser definido com tamanho fixo. O campo “Verificação de Erro” também foi retirado porque o IPv6 supõe que esta função é executada por outras camadas, não tornando o processo redundante. O campo “Tipo de Serviço” foi alterado para “Classe de Tráfego”, e era comumente usado para a marcação de pacotes a fim de determinar uma prioridade de envio. O único campo adicionado no cabeçalho IPv6 foi a “Identificação de Fluxo”, este foi criado para associar vários pacotes da mesma natureza em um único fluxo para fins de classificação e filtragem do tráfego, o que é útil para uma melhor Qualidade de Serviço (*QoS*).

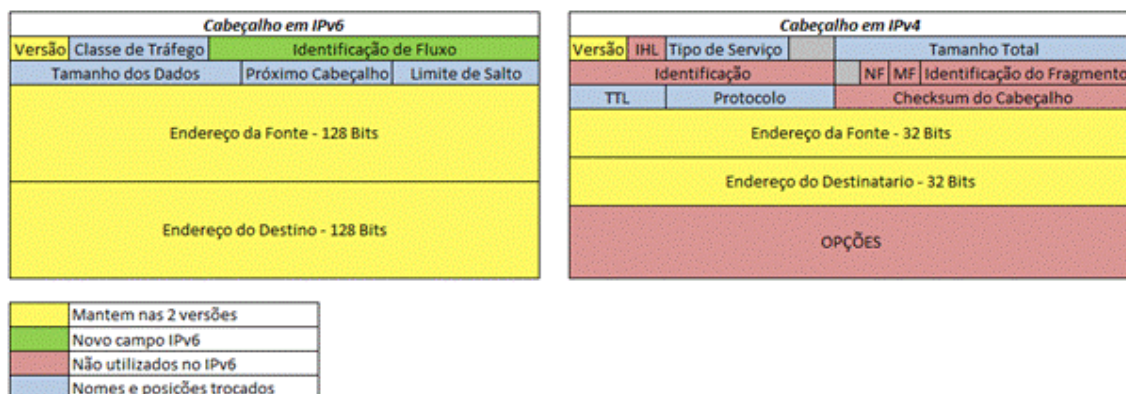


Figura 2.3: Cabeçalhos do IP na versão 6 e 4 e suas alterações.

Fonte: Canno (2013)

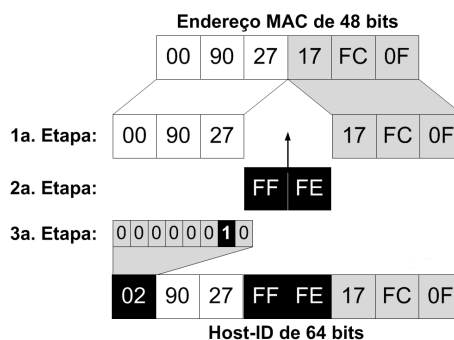


Em relação à segurança atribuída pela utilização do IPSec nativo, é uma técnica que pode ser utilizada diretamente nos nós mas também em dispositivos como roteadores e em *firewalls*. Entre as diversas tecnologias que compõe o IPSec, dois protocolos principais oferecem uma maior flexibilidade, o AH (*Authentication Header*) e ESP (*Encapsulating Security Payload*). Portanto, são possíveis dois modos de atuação: transporte e túnel (Brito, 2013).

1. Transporte - Apenas os dados que fazem parte do *payload* são criptografados, sendo assim, os dados contidos nos cabeçalhos de rede não são alterados, permitindo o roteamento normal pela Internet. Este modo é o mais conveniente para ser utilizado em redes de comunicação ponto-a-ponto;
2. Túnel - Diferentemente do modo transporte, todo o campo IPv6 é criptografado, sendo que para tornar possível roteá-lo pela Internet, é necessário utilizar um túnel, ou seja, reencapsular o pacote em um novo cabeçalho.

No mecanismo de autoconfiguração dos endereços não é necessário nenhum serviço DHCP (*Dynamic Host Configuration Protocol*) (Droms and University, 1997) ativo e de intervenção manual, mesmo assim, é atribuído automaticamente identificadores nos *hosts* da rede. Esse método denominado *Stateless Address Autoconfiguration*, também conhecido como SLAAC, não deixa salvo nenhum registro de endereço atribuído e podem deixar os roteadores menos complexos, mantendo apenas suas configurações mínimas.

A autoconfiguração dos endereços em redes baseadas no IPv6 consiste em duas etapas. Na primeira etapa, para os nós que utilizam o endereço IPv6 completo, os primeiros 64 bits (prefixo) são preenchidos através das mensagens ICMPv6 (Conta et al., 2006) Tipo 134 anunciadas pelo roteador. A segunda etapa de configuração, ou seja, dos outros 64 bits, é a do sufixo. Estes são gerados através do endereço de camada física da interface de rede, o MAC. Neste caso, o MAC tem apenas 48 bits, faltando 16 para completar os 64 bits do sufixo. Para isso é utilizado o EUI-64, uma função de expansão onde é inserido dois octetos padrão (em hexadecimal FFFE) entre os primeiros 32 bits e os últimos. Para finalizar a identificação do sufixo do *host* ainda é adicionado 1 no segundo *bit* menos representativo do primeiro octeto. Este algoritmo de expansão pode ser entendido de forma ilustrada na Figura 2.4.



**Figura 2.4:** Processo de autoconfiguração de nó utilizando o EUI-64.

Fonte: Brito (2013)

Dada uma explicação básica a cerca da utilização do protocolo IP no projeto, serão abordados agora explanações sobre uma das bases que fortalece a IoT.

## 2.3 Redes de Sensores Sem Fio

### 2.3.1 Padrão 802.15.4

O IEEE 802.15.4 é um padrão que vem sendo desenvolvido pelo Grupo de Trabalho 802.15 *Personal Area Network* (PAN) do *Institute of Electrical and Electronics Engineers* (IEEE). É um padrão que define o protocolo e interconexão de dispositivos via rádio frequência de baixo poder de processamento e baixa capacidade de armazenamento (Gutierrez, 2011). Ele tem uma taxa máxima de dados de 250b/s e um potência máxima de saída de 1 mW (Vasseur and Dunkels, 2010). Suas especificações são para redes que usam módulos transmissores e receptores (*transceivers*) de baixa complexidade e custo, tornando a IEEE 802.15.4 uma solução atrativa para os objetos inteligentes.

Várias são as empresas que estão utilizando esta norma para desenvolver protocolos de comunicação proprietários, tais como a ZigBee Alliance (ZigBee®) e HART Communication Foundation (WirelessHART®). Porém, também existem tecnologias de desenvolvimento livres, baseada neste padrão, que podem ser implantadas. Devido à capacidade do IEEE 802.15.4, da viabilidade dos rádios *transceivers* e da liberdade de protocolos de comunicação de código aberto, é possível desenvolver uma diversidade de aplicações no contexto de IoT.

Pelo motivo do padrão IEEE 802.15.4 entender que os dispositivos são de baixa taxa de dados, o tamanho dos pacotes é pequeno, 127 *bytes*. Ela é tipicamente implementada em uma combinação de *hardware* e *software*. A camada mais baixa é formada por *hardware* e o mais alto nível da camada (camada de enlace) são implementadas com *software*.

Para estar de acordo com as especificações, o sistema deve ser formado por dois tipos de componentes básicos: o FFD, do acrônimo *Full-Function Device*, e RFD, do acrônimo *Reduced-Function Device*. O FFD é capaz de desempenhar funções de coordenador da PAN, já o RFD são nós com funções reduzidas que executam tarefas simples. Em termos de funcionamento, o FFD pode operar de três modos: coordenador da rede, coordenador local ou RFD.

Dois ou mais dispositivos dentro de uma rede comunicando no mesmo canal físico formam uma PAN. Entretanto, a rede deve incluir um nó do tipo FFD, que opera como um coordenador da PAN. As topologias possíveis de serem elaboradas são duas: ponto a ponto (*peer-to-peer*) e estrela. Para topologia de rede *peer-to-peer* também deve existir um coordenador, no entanto difere da topologia estrela porque um dispositivo pode se comunicar com dispositivos fora de seu alcance (Gutierrez, 2011), utilizando troca de mensagens com FFD intermediários. Vale salientar que uma rede disposta em comunicação ponto a ponto também pode ser implementada a topologia mesh (malha), onde todos os pontos têm comunicação com qualquer outro ponto. De forma ilustrada, as topologias são mostradas na Figura 2.5.

De acordo com o padrão, são especificados dois tipos de camada:

1. Física: especifica como as mensagens trafegam sobre a camada física do rádio.
2. Enlace: especifica como as mensagens oriundas da camada física são manipuladas.

A camada física é responsável pela transmissão e recepção de *bits* nos módulos de rádio frequência e por controlar o serviço em baixo nível. Ela fornece três bandas de operação: 868-868.6MHz (Europa), 902-928MHz (América do Norte) ou 2400-2483.5MHz (Universal). Esta última, de 2.4GHz, compartilha sua rádio frequência com o 802.11 (Kraemer, 2007) e tem uma considerável sobreposição com esse canal. Porém, devido ao IEEE 802.11 possuir uma alta significância no poder de saída, ele distorce o tráfego do

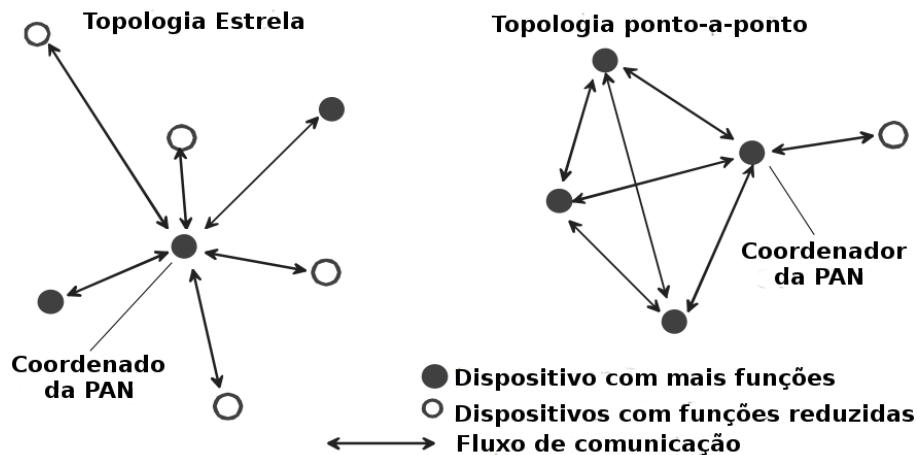


Figura 2.5: Exemplos de topologias para redes 802.15.4.

Fonte: Adaptado [Gutierrez \(2011\)](#)

802.15.4 ([Caldeira, 2012](#)).

As bandas de frequência 868MHz e 915MHz oferecem algumas vantagens em relação à 2.4GHz, por exemplo, necessita de um menor número de dispositivos por área (possuem maior cobertura). Apesar disso, o 2.4GHz é amplamente adotado por uma diversidade de razões, são elas: disponibilidade mundial para uso sem licença, maior taxa de dados (250kbps), mais canais, menor latência e menor consumo de energia.

Em relação a espaços para evitar interferências, 2,4GHz dispõe de dezesseis canais com 5MHz de guarda entre os canais, indo de 2400MHz a 2483,5MHz. Na banda de 868MHz é disposto apenas um canal, este de 20Kbps e, na banda de 915MHz, dez canais com taxa de transferência máxima de 40Kbps e 2MHz de guarda entre cada canal ([Caldeira, 2012](#)).

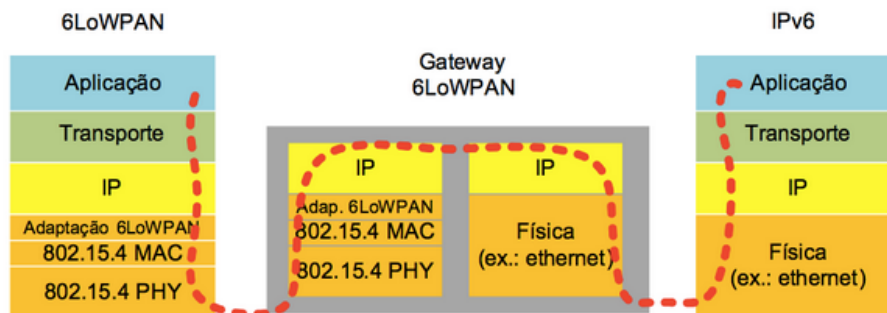
A camada de enlace (camada MAC - *Media Access Control*) fornece dois serviços importantes: o serviço de dados e o de controle de acesso ao rádio (MAC-RL). Este controle de acesso dá-se através da validação dos *frames* de entrada e reconhecimento dos *frames* recebidos. Opcionalmente, a camada MAC oferece um mecanismo de Acesso Múltiplo Dividido pelo Tempo (*time-division multiple access* - TDMA), onde o coordenador da PAN atribui intervalos de tempo aos outros dispositivos da rede e impõe um escalonamento através de transmissões de mensagens de sinalização ([Vasseur and Dunkels, 2010](#)). Portanto, a camada propõe suporte para dois modos distintos: com sinalização (*beacon-enabled*) e sem sinalização (*non beacon-enabled*). Em ambos, para o controle de acesso ao meio é utilizado o mecanismo de CSMA/CA (*Carrier Sense Multiple Access/Collision Avoidance*). No modo *beacon-enabled*, existe um período de contenção que é iniciado quando um *superframe* (*beacon*), definido pelo coordenador, chega até os nós. No outro modo este *beacon* não faz parte do processo de controle de acesso.

Depois de abordar as características e padrões das redes 801.15.4, foi planejado um método que possibilita habilitar em uma rede de sensores o padrão IPV6. Assim, reduzindo os problemas de interoperabilidade entre diferentes sistemas de comunicação.

### 2.3.2 6LoWPAN

De acordo com a RFC1919 ([Kushalnagar et al., 2007](#)), LoWPAN (*Low-Power Wireless Personal Area Network*) é uma rede pessoal sem fio com baixa capacidade de poder computacional. Essas redes são compostas por dispositivos caracterizados com baixo alcance, baixa transmissão, baixo consumo de energia e baixo custo. Para esta norma, foi formado o grupo de trabalho LoWPAN da *Internet Engineering Task Force* (IETF). Nele, dentre outras coisas, são discutidos questões exclusivas de extensões do IPv6 para as LoWPANs. Neste grupo também foi verificada a existência de problemas práticos na RFC4944 ([Montenegro et al., 2007](#)) que trata sobre transmissão de pacotes IPv6 em redes IEEE 802.15.4, portanto existem outros padrões que atualizam esta, são eles: a RFC6282 ([Hui et al., 2011](#)) e RFC6775 ([Shelby et al., 2012](#)). Além destes documentos que especificam o funcionamento e utilização das LoWPANs, os dois trabalhos de E. Kim, a RFC6568 ([Kim et al., 2012a](#)) e a RFC6606 ([Kim et al., 2012b](#)), foram criados para contribuir de forma conceitual e informacional acerca de planejamento, aplicação, desafios e requisitos de IPv6 sobre LoWPAN (6LoWPAN).

Essas redes possuem sérias limitações nas taxas de transmissão e Tamanho Máximo do Pacote (MTU - *Maximum Transmission Unit*). Sendo assim, diferente do sistema de IP tradicional das demais redes que compõem a Internet. Para manter a interoperabilidade e ter uma rede desfragmentada, é fundamental criar uma camada de adaptação para os serviços IP para a manipulação dos pacotes fragmentados ([Vasseur and Dunkels, 2010](#)), assim gerando compatibilidade com os dispositivos embarcados a nível de protocolo. Além do que, a intenção é ser um padrão aberto, universal, em oposição aos protocolos proprietários existentes. A adaptação da pilha IPv6 no sistema operacional do *gateway* é ilustrado na Figura 2.6.



**Figura 2.6:** Tradução do protocolo a nível de enlace.

Fonte: [Moreiras \(2013\)](#)

Além das melhorias que foram expostas na RFC6282, são especificadas também técnicas de compressão. As técnicas LOWPAN\_IPHC (*Internet Protocol Header Compression*) e LOWPAN\_NHC (*Next Header Compression*) afirmam que as técnicas especificadas na RFC4944 não são adequadas na maioria das redes 6LoWPAN. Para atingir o tamanho mínimo de pacote obtido através da compressão de cabeçalho 1 (*Header Compression - HC1*), ambas as partes da comunicação devem adquirir o endereço de enlace local, sendo que ambos os lados da conversa tenham previamente estabelecidos os critérios de formação dos endereços. Nesta técnica, os endereços são limitados ao escopo de redes locais. Porém, um endereço global é fundamental para a comunicação direta através do cliente. Um maior

detalhamento na comparação entre as técnicas podem ser vistas em [Schrickte \(2013\)](#). Sendo assim, a utilização do LOWPAN-IPHC atende aos objetivos deste projeto mencionados na Seção 1.1.

### 2.3.3 Trabalhos relacionados

Vários trabalhos estão relatados na literatura, estes focados no desenvolvimento e planejamento de uma arquitetura de *gateway*, sendo cada um com características particulares. [Steenkamp et al. \(2009\)](#) criou um sistema onde o nó que atua como o *gateway* é configurado remotamente e recupera os dados da rede. [Dun-fan et al. \(2009\)](#) propôs uma arquitetura de *gateway* para aplicações em ambientes monitorados que armazena dados da rede de sensores sem fio e os torna acessíveis através de serviços Web. Estes trabalhos focam em suas aplicações, então não investigam os detalhes de desenvolvimento e utilização do *gateway*. [Qiu et al. \(2009\)](#), [Zhu et al. \(2010\)](#) e [He et al. \(2009\)](#) planejaram uma arquitetura de rede para troca de dados entre nós sensores (ZigBee) e clientes usando um servidor Web. No entanto, a desvantagem destes dois últimos trabalhos é a implementação baseada em protocolos proprietários, tornando inviável e custoso traduzir, no *gateway*, a pilha de protocolo ZigBee com a pilha TCP/IP ([Rao.S et al., 2009](#)).



## Capítulo 3

# Desenvolvimento

*“Laws that oppress people have no moral authority..”  
Richard Stallman*

### 3.1 Procedimentos Metodológicos

A Internet das Coisas tem despertado o interesse não somente nos entusiastas deste novo paradigma, mas também dos investidores, que estimam lucros da ordem de bilhões de dólares já para os próximos anos. Para este trabalho não foi diferente, teve início com o interesse de entender como está organizada a grande rede, a Internet. Porém, no decorrer das pesquisas, o entusiasmo de perceber que ela está caminhando para um novo paradigma encorajou a contribuir para este meio.

A partir daí, deu-se início a uma série de pensamentos e indagações na tentativa de como contribuir, com base nos princípios fundamentais da Internet, onde a complexidade de funcionamento da rede deve estar nas suas bordas, para que este novo paradigma cresça de acordo com o que foi discutido desde os primórdios da comunicação entre máquinas. Porém, devido ser uma área muito extensa e que não existem muitas implementações a cerca da integração dos sistemas com foco em IoT, foi determinado que o escopo seria adicionar aos serviços dos Sistemas de Informação uma grande oferta de meios de manter-se conectado com o ambiente e, assim, poder acessar serviços, interagir com pessoas e outros dispositivos.

Tendo o escopo delimitado, começou-se a pensar e refletir vários meios de obter os objetivos propostos por este trabalho. Então, iniciou-se, uma investigação em busca do estado da arte no uso de *gateways* para a Internet das Coisas. Sendo assim, foi adquirido vários trabalhos que contemplam o assunto, com o objetivo de analisar e comparar a existências de sistemas com o mesmo propósito e suas particularidades. Além disso, foram pesquisadas ferramentas que auxiliem no desenvolvimento da plataforma.

Para o desenvolvimento de um módulo simples de comunicação, será utilizado a linguagem de programação C ([Kernighan and Ritchie, 2010](#)), que possui uma característica simples no que diz respeito ao consumo de memória nos dispositivos embarcados. Pensa-se em construir dois ambientes de desenvolvimento para termos de análise de desempenho, ou seja, criação de protótipo de sistema embarcado com equipamentos diferentes, por exemplo, compiladores, microcontroladores ou programadores. Porém, em testes iniciais pode-se determinar já uma plataforma de desenvolvimento viável para o decorrer do projeto. Afim de criar um ambiente de elaboração do projeto para a comunicação de dados

a serem encaminhados pelo nó intermediador para um nó externo (na Internet), será necessário a criação de, pelo menos, dois módulos comunicadores que, capturados através dos sensores, enviarão dados para este ponto de saída da rede.

Tendo todo um ambiente configurado para a elaboração do projeto, é iniciado a implementação dos protocolos necessários nos sistemas embarcados. Logo após, será integrado no *gateway* uma antena onde esta terá a responsabilidade de transmitir os dados para os nós sensores e repassar para o Sistema Operacional instalado no *gateway*. Esses dados ao chegarem, em nível de camada de enlace, no intermediador será traduzido para o pacote completo do IPv6, já que oriundos da rede de sensores, chegará com o formato do protocolo 6LoWPAN. Por fim, sem ser necessário utilizar alguma aplicação na comunicação entre as bordas, esta mensagem será enviada na interface da máquina configurada para uma comunicação IPv6 com a Internet a um destinatário específico. Esta comunicação também será analisada e testada no sentido oposto a este descrito, onde a mensagem sairá primeiramente do nó que está inserido em uma rede IPv6 para um nó atuador inserido na rede 6LoWPAN.

Sendo assim, para a elaboração de todo o projeto, a metodologia a ser seguida pode ser compreendida como um modelo incremental, onde o conhecimento adquirido em uma etapa é utilizado na etapa seguinte. Caso necessário, as atividades podem ser retomadas, e assim, corrigir os possíveis erros e alterar planejamento.

### 3.1.1 Caracterização da pesquisa

Para garantir que uma pesquisa seja reconhecida como sólida e potencialmente relevante, tanto pelo campo acadêmico quanto pela comunidade em geral, ela deve demonstrar que foi desenvolvida com rigor e que é passível de debate e verificação. Dentre vários métodos de pesquisa, foi escolhido a abordagem *design science* para elaboração deste projeto. Tem como objetivo a produção de conhecimento a partir da busca de soluções para problemas do mundo real utilizando como referências técnicas, procedimentos, fundamentos e ferramentas que só ao aplicadas por profissionais da área conseguem alcançar um produto final (Van Aken and Romme, 2009). Fazendo isto, torna-se um meio importante para aproximar a teoria e prática. Como proposta de orientação para a pesquisa, a abordagem *design science* baseia-se em diretrizes que devem ser seguidas a fim de se obter os resultados desejados, sendo elas:

- Estudar o desenvolvimento de um artefato;
- Verificar a relevância do problema;
- Contribuição da pesquisa para a área de conhecimento;
- Pesquisa rigorosa sobre as técnicas a serem empregadas;
- Uso eficiente de recursos;

Assim a forma de contribuição para este trabalho é projetar um artefato no qual resulta em uma aplicação de conhecimento já existente a partir de uma nova visão. Como forma de contribuição, este trabalho aplicou o tipo “projeto do artefato” para o desenvolvimento da pesquisa, já que, segundo as características desse tipo, os conhecimentos para a construção do artefato em questão já existem, sendo implementado de uma forma diferente.

Durante este trabalho, foram realizados as seguintes atividades:



### 3.1.2 Estudo técnico

A realização do estudo técnico foi através de informações obtidas na documentação das tecnologias envolvidas no desenvolvimento deste projeto, sendo algumas delas, Arduino, eletrônica, protocolos de redes e rádio/sensores, bem como em vídeo aulas, livros ou tutoriais.

### 3.1.3 Desenvolvimento da plataforma

Após o estudo técnico foi elaborado o plano de desenvolvimento da ferramenta de intercomunicação entre rádios sensores e pontos da Internet. As etapas de execução do projeto pode ser visto a seguir:

- Análise de materiais utilizados;
- Modos de comunicação e protocolos entre os rádios comunicadores;
- Configuração de ambiente de desenvolvimento e dispositivos;
- Desenvolvimento de sensores que compõe a rede;
- Planejamento de arquitetura de *gateway* e de rede;
- Implementação de serviço de tradução entre os pontos;

## 3.2 Visão geral

Como o objetivo deste trabalho é inserir objetos na Internet, tornando-a única e uniforme, exige que estes possuam interfaces de comunicação e software adequados para tal. Neste trabalho foram projetados e implementados dispositivos modulares que permitem que equipamentos possam ser integrados à Internet de forma simples e direta. Módulos de sensores com comunicação sem fio adicionado a objetos e um *gateway* foram desenvolvidos ao longo deste.

Inicialmente foi criado nós de redes de sensores, porém, uma rede sem fio não foi implementada sem comunicação com dispositivos externos (dispositivos da Internet). O objetivo de tal implementação é a validação dos dispositivos sem fio, tanto em termos de hardware como de software, sem a influência do *gateway* e de redes externas. Além de permitir o foco no desenvolvimento dos módulos e a validação do hardware, facilitando a detecção de possíveis erros, a decisão por implementar primeiro uma rede local com os nós permitiu que o autor se familiarizasse com os dispositivos antes da integração com outras redes. A própria rede isolada por si só, abre inúmeras portas para o desenvolvimento de aplicações de alto nível. Podendo fazer experimentos de desenvolvimento de serviços que atuem a nível da camada de aplicação da pilha de protocolo utilizada nos sensores de comunicação.

Assim, o tema central do trabalho baseia-se, exclusivamente, no desenvolvimento de uma plataforma de interconexão a ser utilizada por dispositivos das redes de sensores com pontos da Internet. O dispositivo final será capaz de executar tradução de um pacote no protocolo IP para um id que identifica um nó da rede privada de sensores.

## 3.3 Planejamento

Tendo em vista todas as funcionalidades propostas para o projeto, foram feitas algumas restrições aos requisitos mínimos do sistema, no que diz respeito à segurança dos

dados transmitidos pelo dispositivo, a taxa mínima de transmissão dos pacotes de dados, ao valor máximo de memória disponível para o armazenamento de dados históricos, ao tamanho máximo do pacote de dados a ser transmitido, e ao número máximo de dispositivos suportados pela rede ao mesmo tempo.

Assim, inicialmente serão examinados os requisitos do hardware de comunicação RF (rádio frequência). A partir desta definição de hardware, será possível propor um sistema de comunicação eficaz, capaz de ser implementado utilizando a totalidade de recursos computacionais existentes com a plataforma escolhida.

### 3.3.1 Hardware

No projeto dos dispositivos foram inicialmente definidos os requisitos desejados, em alto nível, do ponto de vista do que se deseja para o produto final, sem preocupação com tecnologias utilizadas. Foram avaliadas diversas arquiteturas e configurações de produtos disponíveis no mercado antes de inferir qual das plataformas se enquadraria nos requisitos mínimos do sistema a ser desenvolvido. Desta forma, dentro das soluções existentes, seria ideal utilizar componentes com performance satisfatória e compatível com as tarefas de objetos inteligentes, além do baixo consumo de energia, visto que é um equipamento com o intuito de aumentar a gama de conectividade, e com custo médio de mercado que não inviabilize sua adoção e utilização em massa. De fato, este tipo de tecnologia deve, no mínimo, ter seu retorno financeiro garantido aos usuários, para que os estimulem a adotarem como solução.

Sendo assim, temos como requisito: Utilizar padrão de comunicação sem fio aberto e compatível com a Internet; Material com custo financeiro relativamente barato; Fácil manutenção e flexibilidade de instalação;

- Atmega368PU: São um microcontrolador baseado em RISC AVR com 32 KB de memória flash e recursos de leitura e escrita. Faz parte de uma linha simples de microcontroladores de 8 bits com baixo consumo de energia. Foi selecionado devido suas características de encapsulamento reduzido, ou seja, permite a construção de um rádio pequeno para ser utilizado em conjunto com outros produtos de tamanho reduzido. Outro ponto que importante que o fez ser utilizado no projeto é a disponibilidade da própria fabricante de bibliotecas de código aberto para projetos de recursos reduzidos;
- PIC16x: É uma família de microcontroladores fabricados pela Microchip Technology, que processam dados de 16 bits. Possuem uma capacidade de processamento alto devido a arquitetura Harvard e conjunto de instruções RISC. Assim, produzem códigos mais enxutos sendo importante para para aplicações em Sistemas Embarcados;
- Arduino: É uma plataforma de prototipagem de código aberto baseado em hardware e software de fácil utilização. Placas Arduino são capazes de ler entradas (a luz em um sensor ou um dedo em um botão) e transformá-lo em uma saída (ativação de um motor, ligar um LED...). Devido a experiência de utilização simples e acessível, torna o uso fácil para usuários novatos, como também flexível para usuários avançados. Todo seu código é disponível para que programadores mais experientes possam adaptá-los a seu projeto, contribuir ou detalhar nos quesitos técnicos de funcionamento da plataforma;
- nRF24L01: Este módulo de comunicação utiliza o chip NRF24L01+ fabricado pela Nordic, trabalha na frequência de 2.4 Ghz e possui uma antena embutida e conector

de 10 pinos muito próximos, o que faz deste um dos módulos mais compactos do mercado. No modo transceiver, combina um transmissor e um receptor utilizando componentes de circuito comuns para ambas funções em um só aparelho, ou seja, no mesmo módulo, irá transmitir e receber mensagens utilizando os mesmos recursos para ambas as funcionalidades;

- XBee MaxStream: É um módulo comunicador baseado na pilha de protocolo ZigBee desenvolvido por uma das empresas membro da ZigBee Alliance. Dentre os três tipos de rádios da MaxStream utilizamos a Whip Antenna, nela encontra-se uma antena, com dimensão de aproximadamente 2,5 cm. Com ela, é possível direcionar o feixe de sinal, melhorando o desempenho da rede ([Messias, 2015](#)). Oferece suporte ponto-a-ponto e multiponto com área de cobertura de 120m. 8 pinos de entrada e saída de dados e encriptação limitada a 128 bits;
- Programadores: Dispositivos utilizados para gravar o código nos microcontroladores. Três programadores foram utilizados, são eles: PicBurner (para o PIC), USBtinyISP e um DAPA, este desenvolvido no decorrer do trabalho devido a ausência temporária de outro programador. Ambos os programadores permitem gravar In-Circuit, ou seja, escreve o código no microcontrolador na própria placa de aplicação, reduzindo o tempo e tornando mais prático o desenvolvimento.

### 3.3.2 Software

Após a escolha dos dispositivos de *hardware* a serem utilizados, torna-se necessária a execução de uma metodologia para avaliar, baseado nas limitações do dispositivo, quais critérios serão fundamentais, dentro da arquitetura, para a criação de uma plataforma de *software* capaz de suportar a complexidade do projeto e seus posteriores aprimoramentos. Assim, foram avaliadas as tecnologias e protocolos disponíveis que atendem os requisitos desejados.

- ZigBee: É um protocolo de comunicação voltado à aplicações de baixo consumo e baixa taxa de transferência. Ele é baseado na IEEE 802.15.4, especificando as camadas acima das definidas neste padrão. É bastante utilizado em aplicações de automação residencial, desenvolvido pela ZigBee Alliance e necessita de um pagamento, incluso na compra, para que um dispositivo seja considerado compatível pela organização ([Schrickte, 2013](#));
- ShockBurst: É um protocolo básico que suporta comunicação bidirecional de pacote de dados, incluindo buffer de pacotes, reconhecimento de pacotes e retransmissão automática de pacotes perdidos ([Nordic, 2015](#)). Foi incorporado a *hardware* da série nRF24Lxx desenvolvido pela Nordic Semiconductor;
- EtherCard: Driver para o chip ENC28J60 de Ethernet. Foi utilizado para habilitar comunicação entre o hardware do chip e implementações de roteamento e captura de pacotes. Além de realizar esta tarefa, o EtherCard disponibiliza uma série de implementações relacionado a protocolos e serviços de rede, como por exemplo, implementação da pilha tcp/ip e o serviço de DHCP ([Socher, 2015](#)). Este driver é compatível com o Arduino e, a partir dele, foi elaborado todo o algoritmo de tradução e roteamento entre o cabeçalho IP e identificadores dos rádios comunicadores das redes de sensores. Possui uma licença GPLv2, portanto é livre o uso e alterações necessárias por qualquer indivíduo que deseje executá-lo;
- Firmware de rádio: Dentre os *firmwares* utilizados no projeto, o XBee foi o único que necessitamos de fazer alterações. Nele, identificamos quem tem o papel de

atuar como coordenadores, roteador ou pontos finais. Neste projeto, foi configurado um XBee coordenador, no qual está associado ao *gateway* e um XBee roteador, um dispositivo que se comunica com os demais rádios com sensores dispostos no ambiente. Outra característica que é atribuído ao firmware do XBee é o identificador da rede (PAN ID), através dele identificamos os rádios que fazem parte da rede no qual o único coordenador é o responsável.

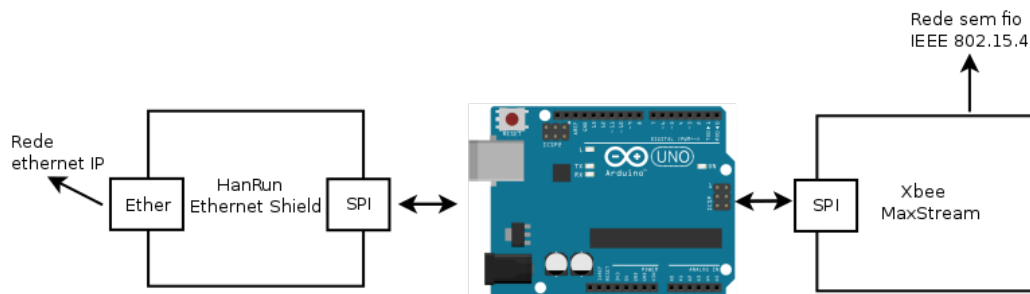
- Pilha de protocolo TCP/IP: Otimizada para dispositivos com pouco poder de processamento e pouca memória. A implementação TCP deste projeto utiliza algumas otimizações de tamanho que são válidos apenas se todos os dados puderem ser enviados em um único pacote. Isto devido, ser um protocolo rodando em cima de dispositivos de redes de sensores. Contudo, não é uma limitação para um microcontrolador, pois as páginas web que são carregadas não possuem dados grandes;
- Implementação da camada de tradução de IP para ID: Implementação da camada de adaptação do IP/ID, incluindo algoritmos de compressão de cabeçalho, conversão de endereços e fragmentação;

### 3.4 Gateway

Uma rede de sensores sem fio não tem seu potencial totalmente explorado caso a rede permaneça isolado, apesar de implementar e usar serviços que a pilha de protocolo TCP/IP fornece. As vantagens de utilizar este padrão se tornam menos significantes ainda, se não há comunicação direta com outros dispositivos IP fora do enlace local. O uso de padrões fechados e mais simples novamente se torna uma opção interessante nestes casos.

A integração dos dispositivos na WPAN a uma rede IP convencional é ponto chave do conceito de Internet das Coisas. Os dispositivos na rede de sensores, apesar de sua simplicidade característica, passam a ser acessíveis de qualquer tipo de rede compatível com IP e da própria Internet.

Algumas diretrizes foram estabelecidas na etapa de planejamento do *gateway* no sentido de manter o dispositivo simples sem comprometer sua funcionalidade e utilidade para os testes pretendidos. Uma das principais diretrizes diz respeito à simplicidade e implementação do hardware. Decidiu-se que o *gateway* deve ser composto por um Arduino, e, associado a ele, um *shield ethernet* HanRun comunicando com a Internet e no outro ponto, um rádio XBee comunicando com a rede de sensores.



**Figura 3.1:** Arquitetura de *Gateway*.

Fonte: elaborada pelo autor.

Portanto, as próximas Seções tratarão dos dispositivos ou *softwares* utilizados no de-

envolvimento do *gateway*. As principais configurações de *hardware* serão abordado, como também as implementações e alterações de código que foram necessários para funcionamento correto da plataforma.

### 3.4.1 Hardware

O *gateway* possui duas interfaces de rede, uma compatível com rede Ethernet e outra com as redes de sensores do protocolo IEEE 802.15.4. Portanto, o *gateway* deve fazer com que datagramas sejam trafegados entre as duas redes de modo que mantenha uma comunicação a nível físico e de enlace. Outra responsabilidade da ferramenta é ter como papel uma camada de adaptação de IP para ID na interface da rede sem fio. Essa camada de adaptação irá comprimir ou decomprimir os pacotes pertencentes a rede privada. Por último, o *gateway* também tem papel de realizar roteamento entre os pontos da rede, assim os pacotes de dados devem ser encaminhados para a interface apropriada.

Para os objetos inteligentes que atuam na rede foram avaliadas configurações de sistemas utilizando microcontroladores em separado dos chips de rádio (dois circuitos integrados distintos). A intenção desta forma de desenvolvimento é tornar o projeto modular, possibilitando o máximo de combinações possíveis de circuitos comunicadores. Para o *gateway*, os *hardwares* selecionados tiveram como requisitos suportar fluxo de dados, isto devido ser um ponto de grande tráfego de dados, fácil manutenção e baixo custo. Portanto, o módulo *ethernet* utilizado foi o HanRun HR911105A.

Com ele é possível enviar e receber dados a partir de qualquer local que tenha acesso a Internet. Possui forma de comunicação formado pela porta *ethernet* RJ45, apresenta chip controlador ENC28J60, um cristal de 25Mhz e conta com um conector de 10 pinos, o que torna prática a ligação desse módulo à diferentes tipos de microcontrolador ([Microduino, 2015](#)). Sua alimentação é feita com 3.3v e as pinagens de comunicação com o arduino, via SPI, pode ser vista na Tabela 3.1:

Módulo ethernet	Arduino Due
CS	D10
SI	D11
SO	D12
SCK	D13
RESET	RESET
INT	D2
VCC	3V3
GND	GND

**Tabela 3.1:** Pinagem de módulo Ethernet HanRun

Fonte: elaborada pelo autor.

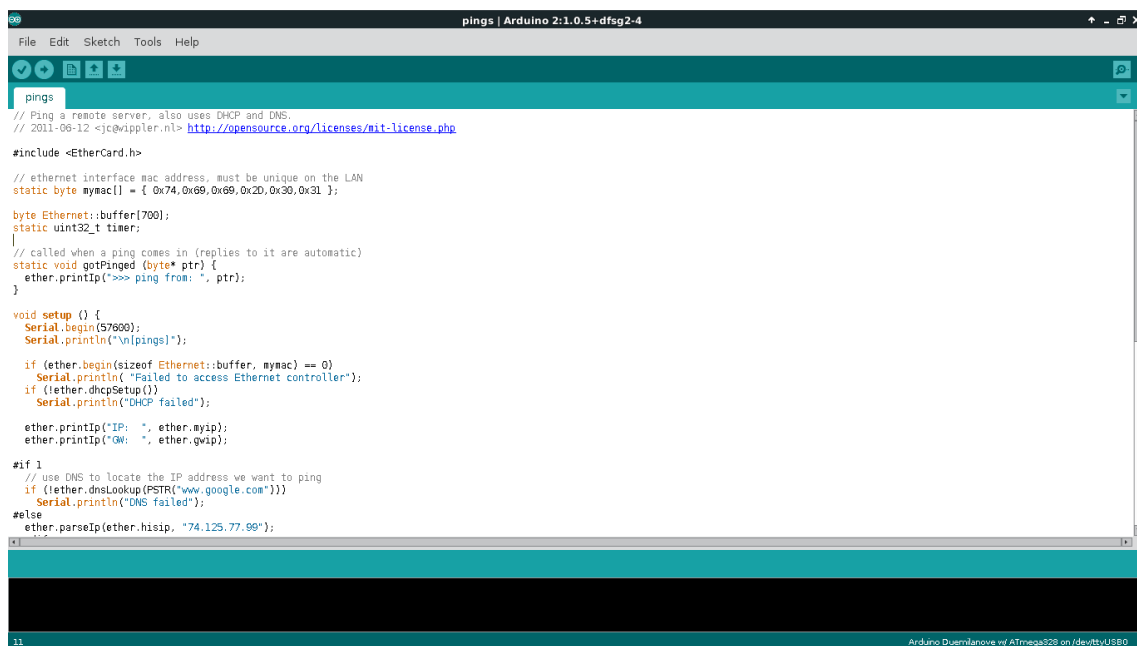
Como visto, os pinos CLKOUT e WOL não são utilizados para esta aplicação.

Outro tipo de *hardware* utilizado no *gateway* foi a plataforma Arduino. Ele, composto por um microcontrolador Atmel, circuitos de entrada/saída e que pode facilmente ser programado via IDE utilizando uma linguagem baseada em C/C++, sem necessariamente ter muitos recursos além de um cabo USB ([Arduino, 2015](#)). Possui uma eletrônica que permite usar a placa com diversas fontes de energia ou até baterias. Objetivando tornar

os estudos de eletrônica mais fáceis, permitiu o acoplamento de circuitos externos através de pinos de conexão em posições padronizadas.

Dentre os tipos de Arduino, o Duemilanove (“2009”) foi o utilizado para o desenvolver do projeto, é uma placa de microcontrolador baseada no ATmega328. Ele possui 14 pinos de entrada/saída digital, 32KB de memória flash 6 entradas analógicas, um cristal oscilador de 16 MHz, uma conexão USB, uma entrada para alimentação e um botão de *reset*.

Uma outra vantagem do Arduino é a interface de programação amigável e fácil comunicação com o microcontrolador. A conexão é feita do computador com o Arduino IDE instalado e o dispositivo via USB, permitindo o *upload* de programas para o controlador. Na Imagem 3.2 é possível enxergar essa interface simples e intuitiva;



**Figura 3.2:** Captura de tela da interface do Arduino IDE.

Fonte: elaborado pelo autor.

Além das funcionalidades principais, também permite um monitoramento da comunicação serial muito útil para entender e controlar o fluxo de mensagem, observação do sistema e até mesmo envio de comandos para uma determinada aplicação.

Como terceiro dispositivo de *hardware* que compõe o *gateway* temos o módulo comunicador XBee. Este sendo o módulo comunicador que fornece interface de comunicação a rede de sensores, possui rede sem fio, com alcance em ambiente externo de 120 metros, e em ambiente interno ou zona urbana 30 metros. Sua frequência de operação é 2.4 GHz e com taxa de dados de 250.000 bps (Sparkfun, 2015). A alimentação deste módulo foi através de uma fonte externa de alimentação de 3.3V. A comunicação com o Arduino é, também, via interface SPI e as configurações di pinos com o arduino pode ser vista na Tabela 3.2.

O desenho gráfico da simulação do circuito construído pode ser visto na Figura 3.3:

XBee MaxStream	Arduino Due
Vcc	3.3V
Tx	Tx
Rx	Rx
GND	GND

Tabela 3.2: Pinagem de módulo comunicador XBee  
Fonte: elaborada pelo autor.

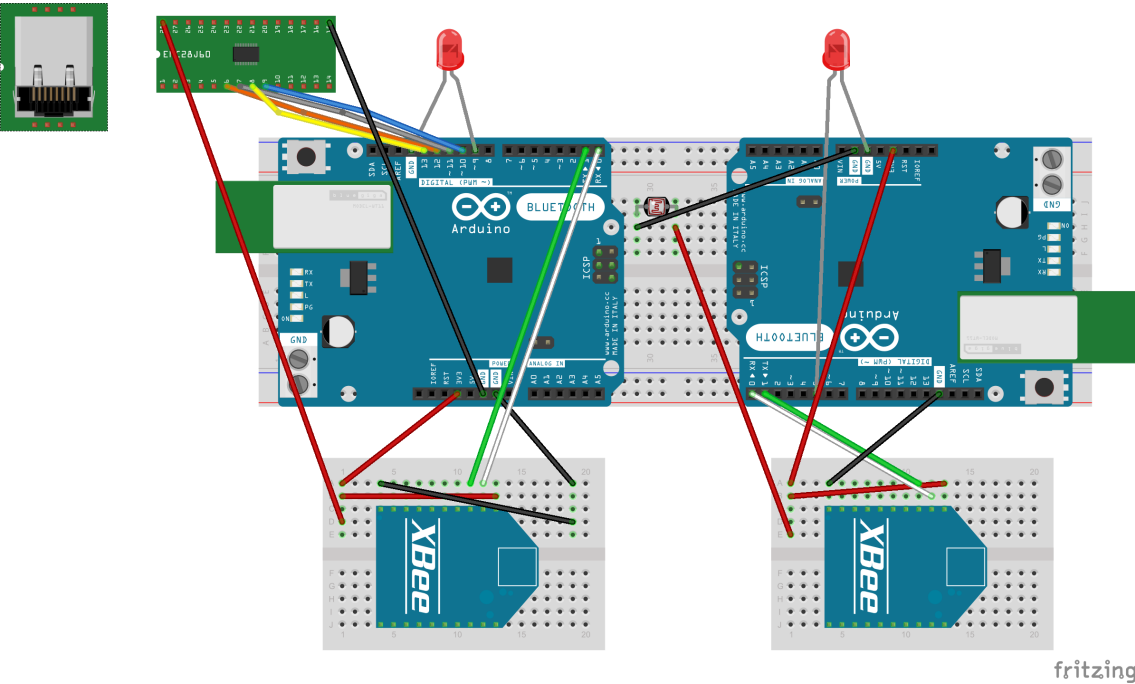


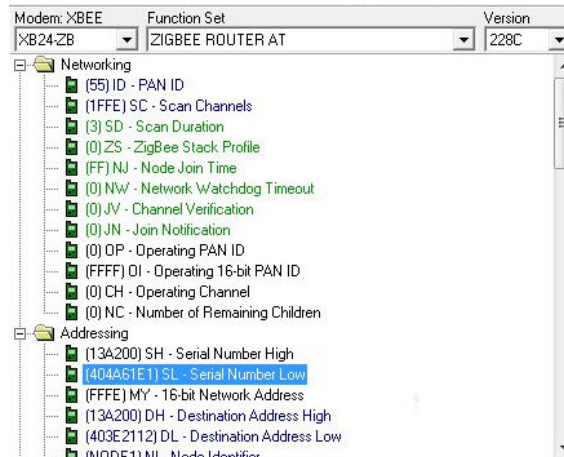
Figura 3.3: Desenho gráfico do gateway e sensor/atuador.  
Fonte: elaborado pelo autor.

3.4.2 Software

Sabendo que é utilizado dois dispositivos com hierarquia diferente na rede, como citado anteriormente, dar início as implementações e configurações de software. Para que os nossos módulos de rádios possam falar uns com os outros é necessário configurá-los de forma que estejam na mesma rede (identificador da PAN, 55) e que utilizem do mesmo protocolo. Portanto, foi atribuído os endereços de identificação nos módulos XBees. Em seguida, é necessário atribuir o endereço do Roteador ao campo de endereço de destino do Coordenador, como também, atribuir endereço do Coordenador no campo de endereço de destino do Roteador, assim, tanto o Roteador sabe quem é seu Coordenador, como o Coordenador sabe para onde encaminhar a mensagem. Os endereços no XBee estão em formato de 64 bits, assim, vêm em duas partes, de baixa prioridade (primeiros 32 bits) e alta prioridade (últimos 32 bits). Os últimos 32 bits dos XBees são comuns a todos os XBees fabricados pela Digi, em 0013A200. Cada módulo de rádio XBee tem o seu próprio e único *Radio Serial Address*. Pode-se ler esse valor usando o software X-CTU.

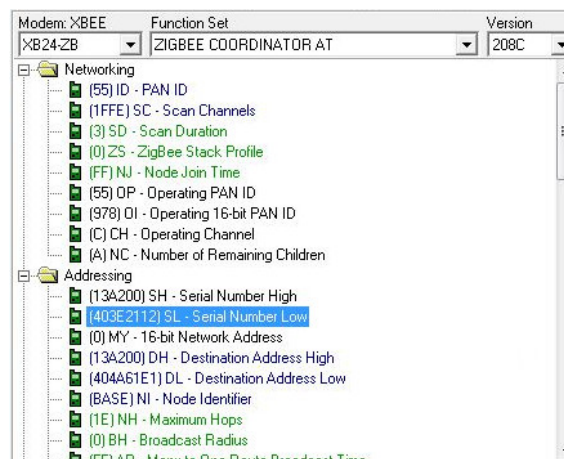
Isso é necessário para saber quando estiver configurando o endereço de destino, para a qual rádio quer conversar.

Nas figuras 3.4 e 3.5, pode-se verificar que os campos em verde são atributos, antes, não configurados ao rádio. Os campos em azul são campos que já pré-configurados, porém o conteúdo do campo foi alterado, por exemplo, o campo do identificador, ambos já existiam, mas o valor passou a ser 55. Na figura também podemos destacar a alteração do campo de endereço de destino com mencionado anteriormente (em azul).



**Figura 3.4:** Captura de tela do XCTU para configuração do Xbee roteador.

Fonte: elaborado pelo autor.



**Figura 3.5:** Captura de tela do XCTU para configuração do Xbee coordenador.

Fonte: elaborado pelo autor.

O próximo passo do projeto em relação a software foi a análise de biblioteca a ser utilizada para a captura de pacotes via ethernet. Portanto, iniciou-se uma busca pelas bibliotecas que atendessem os requisitos. Dentre as pesquisadas, duas se destacaram como prováveis a serem utilizadas, EtherCard e EtherShield. Porém, a EtherCard foi a escolhida devido sua praticidade em carregamentos de página web, mais opções de serviços



implementados e uma documentação mais completa. Dos serviços implementados que a EtherCard possuía a mais em relação a EtherShield são: dhcp, dns, tcpip e udp server. A EtherShield possui apenas uma implementação de serviço web, onde responde requisições udp/tcp em páginas html.

A biblioteca selecionada para a execução do projeto possui a seguinte hierarquia:

- ENC28J60: Esta classe fornecer interface de baixo nível com a interface de rede ENC28J60;
  - EtherCard: Fornece uma interface principal para o rede baseada no dispositivo ENC28J60 e é mais usada pelos usuários;
- Print
  - BufferFiller: Preenche o *buffer* para envio e recepção de dados para a rede;
- StashHeader: Descreve a estrutura de memória utilizada dentro da interface de rede ENC28J60;

Em um dos serviços oferecidos pela biblioteca EtherCard encontramos a possibilidade de implementar a tradução da pilha TCP/IP a um protocolo que os rádios das redes de sensores entendessem. No arquivo *tcpip.c* são implementados métodos que dar a possibilidade de tomar como base para implementar a tradução de protocolos. Os métodos *setMACandIPs()*, *eth\_type\_is\_arp\_and\_my\_ip()* e *eth\_type\_is\_ip\_and\_my\_ip()* descrevem bem os passos que devem ser realizados para fazer uma tradução adequada a nível de rede. Estes dois últimos realizam verificações de tipos de pacotes que estão chegando até ele, assim, analisa se os pacotes são do tipo *arp* ou *ip* e se tem como destino o seu ip. Caso o pacote não tenha seu ip, o mesmo é descartado, entretanto, se possui o ip, é criado um tipo de pacote para resposta. Por seguinte, o método *make\_arp\_answer\_from\_request()* é o responsável por criar este tipo de pacote, também utilizado como base para realizar a ação de resposta para requisições de nós que têm IP atribuído. Baseado neste *driver* foi realizado adaptações e criação de novos serviços para chegar até o objetivo final de tradução.

Como o trabalho é no contexto de Internet das Coisas, uma forma de contribuir ainda mais para esta área foi desenvolver métodos em que nele seriam atribuído funções aos sensores/atuadores da rede. A maneira que utilizamos mais condizente para testar a implementação da tradução de protocolo foi utilizar uma requisição HTTP através de um navegador web. Nele, colocamos o ip do nó atuador, assim o dado o terá como destino e seguirá todos os passos como descrito na Figura 3.6. Junto ao ip do atuador, podemos atribuir dados que o faz realizar ações dependendo do parâmetro estabelecido. Um exemplo de método criado para uma ação do atuador é

```
192.168.0.194/?m=0
```

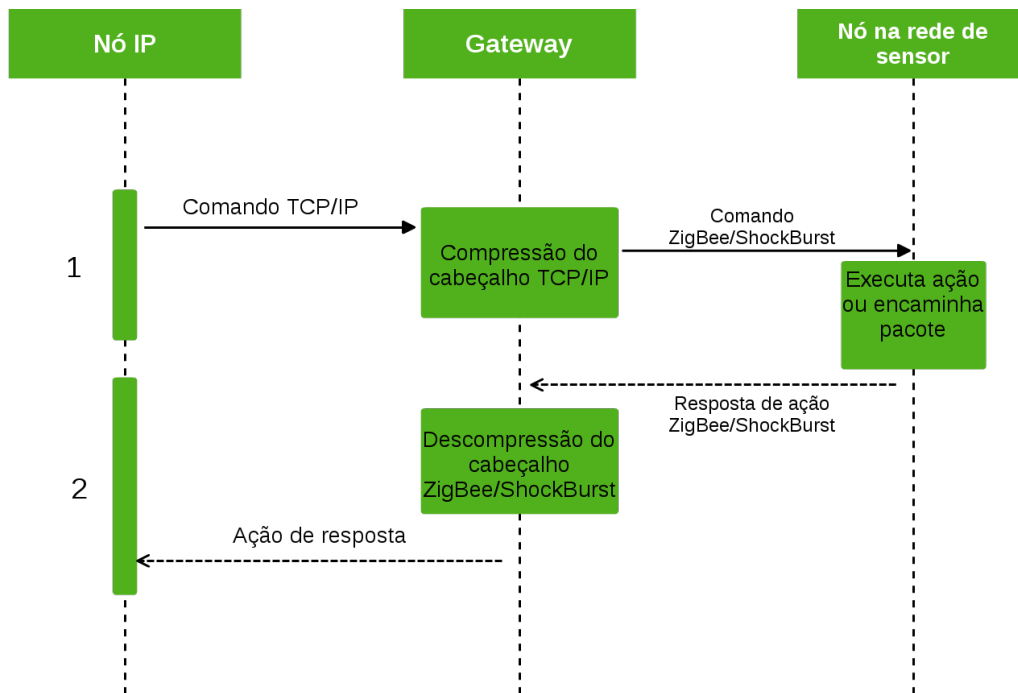
.

Este comando significa dizer que o nó atuador com o ip “192.168.0.194” é o requisitado para atuar. O método “m” é representando um motor de passo que estaria no atuador, assim emulando o movimento de uma, possível, porta de casa. Ao receber o método “m” com o dado “0” significa que o comando é fechar a porte casa de forma remota. Da mesma forma, ao receber o comando

```
192.168.0.194/?m=1
```

significa para abrir a porta. Essa é uma forma de atribuir ao atuador uma ação.

Tendo um ação sobre o nó remoto explicado, outra implementação foi realizada para capturar dados do ambiente, afim de caracteriza-los. Esses dados podem ser adquiridos e



**Figura 3.6:** Fluxo de mensagem entre nó IP, *gateway* e objeto inteligente.

Fonte: elaborado pelo autor.

armazenados em serviços web próprios para a administração de dados para Internet das Coisas usando HTTP, como exemplo, ThingSpeak (Becker et al., 2013). O método criado foi “get” seguido do tipo de dado desejado. Por exemplo, temos a seguinte requisição

192.168.0.194/?get=1

que requisita ao nó com ip “192.168.0.194” a luminosidade (“l”) do ambiente.

Como mostrado, a possibilidade de implementar esses métodos trás uma grandiosidade de utilidade para o contexto de IoT. Isto torna capaz de armazenar uma massa de dados muito grande, assim, fornecendo informações mais relevantes para as tomadas de decisão. Como também, de forma remota, realizar ações sobre objetos ou utensílios em sua casa ou empresa. Sendo uma solução e ponto importante para os problemas e objetivos mencionados no início do trabalho.

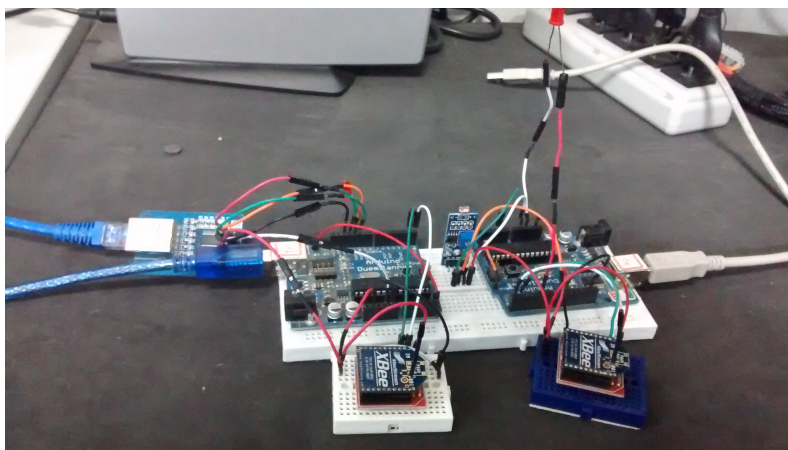
O acesso ao código fonte da tradução de protocolos e código do arduino responsável por realizar o teste e que faz a biblioteca de tradução ser aplicada pode ser visto em [i]

### 3.5 Resultados

O desenvolvimento deste trabalho teve como objetivo principal o desenvolvimento de uma plataforma de interconexão entre redes de sensores e a Internet no contexto de Internet das Coisas, abrindo possibilidade de realizar outros experimentos e estudos a nível de aplicação. Durante a realização do trabalho foi adquirido conhecimentos sobre

[i]Disponível em: <https://denispinheiro@bitbucket.org/denispinheiro/translationprotocol.git>

estruturação e configuração das redes de sensores, implementação de protocolo de comunicação, montagem de circuitos eletrônicos e melhor entendimento da funcionalidade da pilha de protocolo TCP/IP. Essas atividades favoreceram uma experiência fantástica por tratar de uma área muito forte para o mercado de trabalho e assuntos complexos que, ao domina-los, concede ao pesquisador um avanço intelectual bem considerável. Os assuntos voltados para a Internet das Coisas é visto com bastante crédito no momento atual da ciência, são pontos fortemente discutidos no meio tecnológico atualmente. A imagem real do que foi desenvolvido por ser visto através da Figura 3.7.



**Figura 3.7:** Imagem real do *gateway* ao lado do sensor/atuador.

Fonte: elaborado pelo autor.

Então, as atividades e objetivos planejados para este projeto foram desempenhadas da forma mais adequada possível, mostrando, de forma prática, todo o conhecimento adquirido no decorrer de todo o curso de graduação. O protótipo de *gateway* desenvolvido neste trabalho pode ser utilizado por outro plano de projeto para fins experimentais e desenvolvimento de serviços a nível de aplicação. Como também, pode ser usado como base para novas pesquisas sobre o tema ou até coloca-lo a prova de conceito.

Deixa, como fonte de estudo, uma plataforma que insere, no contexto de Internet das Coisas, as redes de sensores e seus respectivos objetos inteligentes a grande rede da Internet. Isso possibilita um ganho enorme de dados que caracteriza o ambiente, tornando mais forte o poder de decisão dos sistemas de informação.

### 3.6 Considerações Finais

Os dispositivos foram projetados e construídos com sucesso, as implementações foram parciais. Testes simples comprovaram seu funcionamento e fez abrir portas para a realização de experimentos mais profundos envolvendo a criação de uma plataforma de interconexão barata e simples. Apesar de serem apenas protótipos, os dispositivos apresentam potencial para uso em aplicações acadêmicas dado o tamanho reduzido dos módulos e a flexibilidade de configuração proporcionada pelo uso dos recursos no *gateway*.



## Capítulo 4

# Conclusões

*“The science of today is the technology of tomorrow.”  
Edward Teller*

Finalmente, como todo trabalho, no decorrer do desenvolvimento foram encontradas algumas dificuldades . Entre elas, o desenvolvimento de novos protocolos de comunicação para dispositivos com baixo poder de processamento e memória. O fato do desenvolvimento para sistemas embarcados ser diferente para o desenvolvimento de sistemas para computadores pessoais dificultou consideravelmente o andamento do trabalho, tendo, algumas vezes, ser necessário alterar o cronograma do projeto. Também, foi encontrado limitação nos dispositivos disponíveis no laboratório que preenchessem as restrições de desenvolvimento. Por isso, alguns objetivos do projeto não foram alcançados.

### 4.1 Trabalhos futuros

No atual estado do trabalho são necessários alguns aprimoramentos que podem ser incluídos em versões futuras do projeto. Um dos pontos principais que pode ser realizado é a tradução do protocolo IP para um protocolo padronizado, como por exemplo o 6LowPAN. Este, oferece vários outros serviços e obviamente seria o avanço enorme utilizar já protocolo IPv6 em redes de sensores, visto que toda a Internet esta voltado para este mesmo objetivo. As vantagens de tornar toda a comunicação entre os dois extremos da rede com um único protocolo padrão pode ter sua leitura retomada na Seção 2.2

Ainda em relação ao tráfego de dados, pode ser analisado a possibilidade de implementar algoritmo de segurança no sistema de tradução de protocolos. A questão de usar criptografia nas PAN's talvez torne o sistema mais seguro, preenchendo um forte ponto e requisito de qualquer sistema de computação.

No que diz respeito a arquitetura de *gateway*, pode ser utilizado, substituindo o Arduino, uma outra plataforma que sustento um poder de processamento maior, visto que o fluxo de dados a ser processado e analisado pelo *gateway* é grande. O fato de ser arduino, pode-se criar um gargalo na rede, tornando o arduino o ponto de vulnerabilidade no quesito disponibilidade e confiabilidade do sistema. Caso o dispositivo responsável pelo processamento e análise de dados seja um computador mais robusto, rodando Linux, talvez, possa tornar um trabalho mais relevante para o cenário de IoT.

Como o projeto pretendeu desenvolver diversos módulos de sensores embarcados, torna-se possível também simular uma rede com um maior número de sensores. Assim, é possível

criar sub-redes distintas, e avaliar a quantidade de perda de dados quando diversos dispositivos estão enviando pacotes simultaneamente na rede. Assim, pode avaliar o algoritmo de roteamento entre os roteadores das redes de sensores, ver se eles estão cumprindo o papel e encaminhando os dados para seus devido destinos

## Referências Bibliográficas

- Arduino (2015), ‘Arduino introduction’. Novembro 2015.  
**URL:** <https://www.arduino.cc/en/Guide/Introduction>  
(Citado na página 25)
- Ashton, K. (2009), ‘That “internet of things” thing: In the real world, things matter more than ideas’, RFID Journal.  
(Citado na página 8)
- Becker, Martin, Juergen Mueller, Andreas Hotho and Gerd Stumme (2013), A generic platform for ubiquitous and subjective data, *em* ‘Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication’, UbiComp ’13 Adjunct, ACM, New York, NY, USA, pp. 1175–1182.  
**URL:** <http://doi.acm.org/10.1145/2494091.2499776>  
(Citado na página 30)
- Borges Neto, J. B., R. M. C. Andrade and P. F. Ribeiro Neto (2010), Wireless Sensor Networks Advances for Ubiquitous Computing, *em* ‘Designing Solutions-Based Ubiquitous and Pervasive Computing: New Issues and Trends’, IGI Global, pp. 175–189.  
(Citado na página 3)
- Brito, Samuel H. B. (2013), *IPv6 - O novo protocolo da Internet*, 1ª edição, Novatec Editora.  
(Citado nas páginas 3, 7, 12, e 13)
- Caldeira, J. F. (2012), Estudo e desenvolvimento de uma plataforma de comunicação sem fio para redes elétricas inteligentes, Dissertação de mestrado, Universidade Federal do Rio de Janeiro.  
(Citado na página 15)
- Campos, B. da S., J. J. P. C. Rodrigues, L. D. P. Mendes, E. F. Nakamura and C. M. S. Figueiredo (2011), Design and construction of a wireless sensor and actuator network gateway based on 6LoWPAN, *em* ‘EUROCON - International Conference on Computer as a Tool (EUROCON), 2011 IEEE’, pp. 27–29.  
(Citado na página 11)
- Canno, R. M. (2013), ‘Redes ip i: Comparativo entre ipv4 e ipv6’. Janeiro 2013.  
**URL:** <http://www.teleco.com.br/tutoriais/tutorialredeip1/default.asp>  
(Citado na página 12)

- Coetzee, L. and J. Eksteen (2011), The Internet of Things - Promise for the Future? An Introduction, *em* 'IST-Africa Conference e Exhibition'.  
(Citado nas páginas 2, 7, e 9)
- Conta, A., Transwitch, S. Deering, Cisco Systems, M. Gupta and Tropes Networks (2006), 'Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification', Internet Engineering Task Force (IETF) RFC 4443.  
**URL:** <https://tools.ietf.org/html/rfc4443>  
(Citado na página 13)
- Deering, S. and R. Hinden (1998), 'Internet Protocol, Version 6 (IPv6) Specification', Internet Engineering Task Force (IETF) RFC 2460 (Draft Standard). Obsoletes RFC 1883. Updated by RFCs 5095, 5722, 5871.  
**URL:** <http://www.ietf.org/rfc/rfc2460.txt>  
(Citado na página 11)
- Diogo, P., L.P. Reis and N. Vasco Lopes (2014), Internet of things: A system's architecture proposal, *em* 'Information Systems and Technologies (CISTI), 2014 9th Iberian Conference on', pp. 1–6.  
(Citado nas páginas 3 e 9)
- Droms, R. and Bucknell University (1997), 'Dynamic Host Configuration Protocol', Internet Engineering Task Force (IETF) RFC 2131.  
**URL:** <https://www.ietf.org/rfc/rfc2131.txt>  
(Citado na página 13)
- Dun-fan, Y., M. Liang-liang and W. Wang (2009), Design and Implementation of Wireless Sensor Network Gateway Based on Environmental Monitoring, *em* 'Environmental Science and Information Application Technology, 2009. ESIAT 2009. International Conference on', pp. 289–292.  
(Citado na página 17)
- Frankel, S., NIST and S. Krishnan (2011), 'IP Security (IPsec) and Internet Key Exchange (IKE) Document Roadmap', Internet Engineering Task Force (IETF) RFC 6071.  
**URL:** <http://www.ietf.org/rfc/rfc6071.txt>  
(Citado na página 12)
- Gomes, D. A. (2010), *Web Services SOAP em Java: Guia Prático para o desenvolvimento de web services em Java*, Novatec.  
(Citado na página 3)
- Gutierrez, J. A. et al (2011), 'Ieee standard for local and metropolitan area networks—part 15.4: Low-rate wireless personal area networks (lr-wpans)', *IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)* pp. 1–314.  
(Citado nas páginas 14 e 15)
- Harpham, B. (2014), 'How canadian organizations use the internet of things'. Setembro 2014.  
**URL:** <http://www.itworldcanada.com/blog/how-canadian-organizations-use-the-internet-of-things/97386>  
(Citado na página 8)



- He, H., Z. Yue and X. Wang (2009), Design and Realization of Wireless Sensor Network Gateway Based on ZigBee and GPRS, *em* 'Information and Computing Science, 2009. ICIC '09. Second International Conference on', pp. 21–22.  
(Citado na página 17)
- Hui, J., Arch Rock Corporation and P. Thubert (2011), 'Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks', Internet Engineering Task Force (IETF) RFC 6282.  
**URL:** <http://www.ietf.org/rfc/rfc6282.txt>  
(Citado na página 16)
- Jayanthi, J.G. and S.A. Rabara (2010), Next generation internet protocol -Technical realms, *em* 'Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on', Vol. 9, pp. 394–399.  
(Citado na página 12)
- Kernighan, B. W and D. M. Ritchie (2010), *The C Programming Language*, 2ª edição, Prentice Hall.  
(Citado na página 19)
- Khan, R., S. U. Khan, R. Zaheer and S. Khan (2009), A gateway solution for ipv6 wireless sensor networks, *em* 'Ultra Modern Telecommunications Workshops, 2009. ICUMT '09. International Conference on', pp. 1–6.  
(Citado na página 2)
- Kim, E., ETRI and D. Kaspar (2012a), 'Design and Application Spaces for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)', Internet Engineering Task Force (IETF) RFC 6568.  
**URL:** <http://www.ietf.org/rfc/rfc6568.txt>  
(Citado na página 16)
- Kim, E., ETRI and D. Kaspar (2012b), 'Problem Statement and Requirements for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)', Internet Engineering Task Force (IETF) RFC 6606.  
**URL:** <http://www.ietf.org/rfc/rfc6606.txt>  
(Citado na página 16)
- Kraemer, B. et al (2007), 'IEEE Standard for Information technology— Telecommunications and information Local and metropolitan area networks', *IEEE Std 802.11-2007* p. 11.  
(Citado na página 14)
- Kushalnagar, N., G. Montenegro and C. Schumacher (2007), 'IPv6 over Low-Power Wireless Personal Area Network (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals', Internet Engineering Task Force (IETF) RFC 4919 (Informational).  
**URL:** <http://www.ietf.org/rfc/rfc4919.txt>  
(Citado na página 16)
- Loureiro, Antonio A. F., et al (2003), Redes de sensores sem fio, *em* 'In Proceedings of the 21st Brazilian Symposium on Computer Networks (SBRC'03)', pp. 179–226.  
(Citado na página 3)

- Medeiros, M. (2006), *Banco de dados para sistemas de informação*, Visual Books.  
(Citado na página 10)
- Messias, Antônio Rogério (2015), ‘Controle remoto e requisições de dados via xbee/zigbee (ieee 802.15.4’. Outubro 2015.  
**URL:** <http://www.rogercom.com/ZigBee/ZigBee.htm>  
(Citado na página 23)
- Microduino (2015), ‘Datasheet harrun’. Novembro 2015.  
**URL:** <https://www.microduino.cc/wiki/images/6/66/HR911105A.PDF>  
(Citado na página 25)
- Montenegro, G., Microsoft Corporation and N. Kushalnagar (2007), ‘Transmission of IPv6 Packets over IEEE 802.15.4 Networks’, Internet Engineering Task Force (IETF) RFC 4944.  
**URL:** <http://www.ietf.org/rfc/rfc4944.txt>  
(Citado na página 16)
- Moreiras, A. M. (2013), ‘Zigbee usa agora 6lowpan! sua próxima lâmpada terá ipv6?’. Março 2013.  
**URL:** <http://ipv6.br/zigbee-usa-agora-6lowpan-sua-proxima-lampada-tera-ipv6/>  
(Citado nas páginas 3 e 16)
- Nordic, S (2015), ‘Datasheet shockburst protocol’. Novembro 2015.  
**URL:** [https://devzone.nordicsemi.com/documentation/nrf51/4.2.0/html/group\\_\\_esb\\_\\_users\\_\\_guide.html](https://devzone.nordicsemi.com/documentation/nrf51/4.2.0/html/group__esb__users__guide.html)  
(Citado na página 23)
- Postel, J. (1981), ‘Internet Protocol’, Internet Engineering Task Force (IETF) RFC 791 (Standard). Obsoletes RFC 760. Updated by RFC 1349.  
**URL:** <http://www.ietf.org/rfc/rfc791.txt>  
(Citado na página 11)
- Qiu, P., U. Heo and J. Choi (2009), The web-sensor gateway architecture for Zigbee, *em* ‘Consumer Electronics, 2009. ISCE ’09. IEEE 13th International Symposium on’, pp. 661–664.  
(Citado na página 17)
- Rao, S., Gopinath, Zeldy Suryady, U. Sarwar and M. Abbas (2009), A gateway solution for ipv6 wireless sensor networks, *em* ‘Ultra Modern Telecommunications Workshops, 2009. ICUMT ’09. International Conference on’, pp. 1–6.  
(Citado nas páginas 10 e 17)
- Santucci, G. (2010), The Internet of Things: Between the Revolution of the Internet and the Metamorphosis of Objects, *em* ‘Vision and Challenges for Realising the Internet of Things’, Publications Office of the European Union, pp. 175–189.  
(Citado nas páginas 8 e 9)
- Schrickte, L. F. (2013), Projeto, implementação e avaliação de desempenho de nós e gateway 6lowpan, Dissertação de mestrado, Universidade Federal de Santa Catarina.  
(Citado nas páginas 12, 17, e 23)

- Shelby, Z., Sensinode and S. Chakrabarti (2012), ‘Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)’, Internet Engineering Task Force (IETF) RFC 6775.  
**URL:** <http://www.ietf.org/rfc/rfc6775.txt>  
(Citado na página 16)
- Socher, G. Stang, P. (2015), ‘Ethercard driver’. Novembro 2015.  
**URL:** <http://jeelabs.net/pub/docs/ethercard/>  
(Citado na página 23)
- Socolofsky, T. and C. Kale (1991), ‘A TCP/IP Tutorial’, Internet Engineering Task Force (IETF) RFC 1180 (Informational).  
**URL:** <https://tools.ietf.org/html/rfc1180>  
(Citado na página 11)
- Sparkfun (2015), ‘Xbee®/xbee-pro rf modules’. Novembro 2015.  
**URL:** <https://www.sparkfun.com/datasheets/Wireless/Zigbee/XBee-Datasheet.pdf>  
(Citado na página 26)
- Steenkamp, L., S. Kaplan and R.H. Wilkinson (2009), Wireless sensor network gateway, *em* ‘AFRICON, 2009. AFRICON ’09.’, pp. 1–6.  
(Citado na página 17)
- Van Aken, J. E. and G. Romme (2009), Reinventing the future: adding design science to the repertoire of organization and management studies, *em* ‘Organization Management Journal’, pp. 5–12.  
(Citado na página 20)
- Vasseur, J. and A. Dunkels (2010), *Interconnecting Smart Objects with IP: The Next Internet*, Morgan Kaufmann Publishers Inc.  
(Citado nas páginas 2, 4, 9, 10, 12, 14, 15, e 16)
- Weiser, Mark (1991), ‘The computer for the 21st century’, pp. 265(3):94–104.  
(Citado nas páginas 1 e 2)
- Zhu, Q, R. Wang, Q. Chen, Y. Liu and W. Qin† (2010), Iot gateway: Bridging wireless sensor networks into internet of things, *em* ‘Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on’, pp. 347–352.  
(Citado na página 17)