

CONTADOR DE PESSOAS POR ANALISADOR DE PACOTES WIFI.

Albano Leo Ehrenbrink

Universidade Federal do Rio de Janeiro, Rua Horácio Macedo, Bloco G, 2030 – 101, Rio de Janeiro, RJ, Brasil
albano@ehrenbrink.com

Abstract. Contadores de pessoas geralmente são usados para entender o comportamento da população ou de consumidores. Também são usados por administradores de locais com muito fluxo de pessoas como parques de diversão, transporte público e congressos para medir o interesse do público e gerenciar recursos para uma melhor experiência. Este trabalho mede a quantidade de endereços e placa de rede sem fio no entorno de um dispositivo de borda que processa localmente a informação de forma a descartar a identificação das pessoas antes de enviar a contagem pelo protocolo de modulação LoRa (Long Range) a um Gateway LoRaWAN distante a até 700 metros. Este Gateway transmite a contagem a um Broker MQTT livre instalado na The Things Network que retransmite a informação ao um painel de informação a um serviço integrado TagIO.

Keywords: Contador de pessoas, SNIFFER, LoRa, Internet das Coisas, MQTT

1. INTRODUÇÃO

Este trabalho tem o intuito de contar pessoas nas proximidades de um ponto de interesse sem identifica-las, analisando os pacotes de requisição de rede dos dispositivos móveis.

Conforme (HAOCHAO LI, 2015), a utilização de portões de controle ou pontos específicos para a contagem de pessoas não é o indicado na maioria das situações sendo a contagem de dispositivos WiFi a mais indicada.

Apesar do uso de técnicas de análise de pacotes conhecido como *SNIFFER* estarem normalmente relacionadas a problemas de privacidade, este trabalho utiliza o conceito de computação na borda para descaracterizar qualquer informação que possa identificar os indivíduos, transmitindo apenas o número de pessoas.

De modo a permitir a instalação do dispositivo de contagem em locais com pouca infraestrutura foi utilizado para comunicação o protocolo de modulação LoRa possuindo capacidade de transmissão suficiente (ALOÿS AUGUSTIN, 2016) para a finalidade deste contador com a utilização do protocolo MQTT - *Message Queuing Telemetry Transport*.

2. FUNDAMENTAÇÃO TEÓRICA

O desenvolvimento deste trabalho utiliza diversos pontos normalmente utilizados em dispositivos de Internet das Coisas e desenvolvidos ao longo do curso CPS 731 do PESC, como:

1. Processamento de borda de baixo consumo (ESP32);
2. Protocolo de modulação de baixo consumo e logo alcance (LoRa);
3. Protocolo de comunicação com baixa exigência de recursos como cabeçalho pequeno (MQTT);
4. Gateway para tradução de um protocolo físico para outro (LoRaWAN);
5. Tradutor de comunicação da postagem MQTT conhecido como Broker MQTT (Node JS rodando no The Things Network);
6. Painel de visualização da informação (no dispositivo de borda e em um painel de informação integrado ao The Things Network).

Em tempo, o The Things Network é um projeto global cuja missão é construir uma rede de Internet das Coisas aberta. (THE THINGS NETWORK, 2016)

2.1 Architeturata

A contagem começa com o sensor, que é a antena do dispositivo de borda, passa pelo processamento na borda para filtrar contagens duplicadas, criar a sentença de comunicação MQTT e descartar antes do envio as informações de identificação dos usuários, transmissão por modulação LoRa, Tradução de LoRa para TCP/IP e de MQTT para HTTP e por fim um painel de visualização, conforme é resumido na Figura 1.

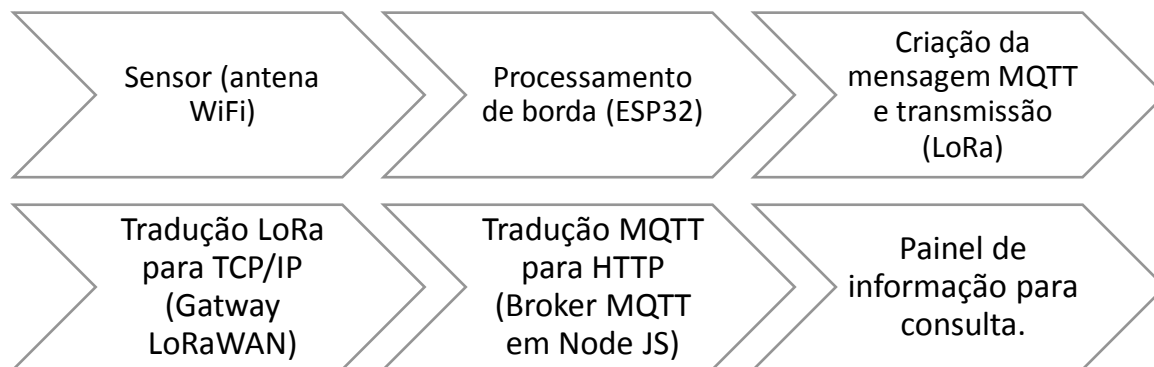


Figura 1 - Arquitetura

2.2 Software

O software instalado no dispositivo é uma modificação do projeto PAXCOUNT desenvolvido inicialmente por Verkehrsrot (VERKEHRSROT, 2018) e posteriormente suportado pela comunidade.

A modificação implementada está disponível em https://github.com/AlbanoL/Sniffer_WiFi_ESP32_LoRa_TTN. Está modificação contempla a configuração da conta The Things Network, implementação da frequência de 433Mhz não disponível no projeto original, e configuração do hardware de borda utilizado (endereço dos pinos e recursos disponíveis).

2.2.1 Instalação

2.2.1.1 Software embarcado no dispositivo de borda ESP32

Instale o Microsoft Visual Studio Code (<https://code.visualstudio.com/>) para Windows, macOS ou Linux, o editor gratuito da Microsoft.

No Visual Studio Code, instale o suporte para PlatformIO. Selecione File / Preferences / Extensions e procure por "PlatformIO". Clique em "Install" na lista de resultados em "Platform IO IDE" e reinicie o Visual Studio Code após a instalação. Pode levar vários minutos para a extensão ser instalada.

Na barra de status na parte inferior da janela, você verá "Instalando o PlatformIO IDE ...". Se uma nova nota for sinalizada à direita do símbolo com a campainha, você poderá abri-la clicando no símbolo. Em seguida, você será solicitado a reiniciar o código do Visual Studio (recarregar agora).

Faça o download do código-fonte do GitHub https://github.com/AlbanoL/Sniffer_WiFi_ESP32_LoRa_TTN como um arquivo zip usando o botão Clonar ou baixar e descompacte-o em qualquer pasta.

Conecte a placa por um cabo USB ao computador e verifique se foi reconhecida no Gerenciador de Dispositivos em Portas COM/LPT.

No Microsoft Visual Studio Code, clique em arquivo / abrir pasta e selecione a pasta onde você extraiu o código do GitHub. Em seguida, os arquivos existentes são exibidos.

No canto inferior esquerdo, você verá vários ícones pequenos no Visual Studio Code.

Clique na pequena lixeira para limpar o código (PlatformIO: Clean).

Quando estiver pronto clique em no símbolo de certo/check (PlatformIO: Build).

Quando estiver pronto clique no símbolo seta para a direita (PlatformIO: Upload).


2.2.1.2 Configurando para outra conta no The Things Network

Caso queira aproveitar o código para outra conta TTN, abra o console do The Things Network <https://console.thethingsnetwork.org/> (menu em seu nome de usuário), clique em Aplicativos e adicione o aplicativo. Em ID do aplicativo, nomeie seu contador e clique em adicionar aplicativo. Você verá a janela Visão Geral do Aplicativo.



Na caixa Dispositivos, clique em registrar dispositivo. Defina qualquer ID de dispositivo e clique nas duas setas cruzadas para o Dispositivo EUI para que um texto cinza apareça no campo: "este campo será gerado". O botão fica verde - depois clique em salvar. Agora o contador é criado como um novo dispositivo.



A Visão geral do dispositivo exibe o dispositivo EUI e o aplicativo EUI. Alterne para a exibição msb clicando no campo com os dois colchetes angulares para a direita e para a esquerda.

Esta informação é necessária para o software no arquivo src/Iraconf.h que precisa ser configurado para interligar seu dispositivo com sua conta TTN.

 **CONSOLE**
COMMUNITY EDITION

ApplicationsGatewaysSupport

 **Albano** 

Applications >  contadordepeessoas > Devices >  **iot1albano**

OverviewDataSettings

DEVICE OVERVIEW

Application ID

contadordepeessoas

Device ID


iot1albano

Activation Method

OTAA


Device EUI

<>↕00 B9 17 F7 94 3B 26 03




Application EUI

<>↕70 B3 D5 7E D0 01 D4 44



App Key

<>↕🔒37 56 0D 47 71 30 10 0E E3 8E 9B DF 14 BA 2C 4C



Status

● *never seen*

Frames up

0 [reset frame counters](#)

Frames down

0

Figura 2 - Endereço do dispositivo no console do The Things Network

```
PIO Home  C loraconf.h  src > C loraconf.h > ...
1  #if(HAS_LORA)
2
3  /*****
4   * LMIC LoRaWAN configuration
5   *
6   * Read the values from TTN console (or whatever applies), insert them here,
7   * and rename this file to src/loraconf.h
8   *
9   * Note that DEVEUI, APPEUI and APPKEY should all be specified in MSB format.
10  * (This is different from standard LMIC-Arduino which expects DEVEUI and APPEUI
11  * in LSB format.)
12
13  * Set your DEVEUI here, if you have one. You can leave this untouched,
14  * then the DEVEUI will be generated during runtime from device's MAC adress
15  * and will be displayed on device's screen as well as on serial console.
16  *
17  * NOTE: Use MSB format (as displayed in TTN console, so you can cut & paste
18  * from there)
19  * For TTN, APPEUI in MSB format always starts with 0x70, 0xB3, 0xD5
20  *
21  * Note: If using a board with Microchip 24AA02E64 Unique ID for deveui,
22  * the DEVEUI will be overwritten by the one contained in the Microchip module
23  *
24  *****/
25
26 static const u1_t DEVEUI[8] = { 0x80, 0xB9, 0x17, 0xF7, 0x94, 0x3B, 0x26, 0xB3 };
27
28 static const u1_t APPEUI[8] = { 0x70, 0xB3, 0xD5, 0x7E, 0x00, 0x01, 0xD4, 0x44 };
29
30 static const u1_t APPKEY[16] = { 0x37, 0x56, 0x8D, 0x47, 0x71, 0x30, 0x10, 0x0E, 0xF3, 0x8E, 0x9B, 0xDF, 0x14, 0xBA, 0x2C, 0x4C };
31
32 #endif // HAS_LORA
```

Figura 3 - Endereço do dispositivo configurado no arquivo loraconf.h copiado do console do The Things Network

2.2.1.3 Broker MQTT e painel de informação

As informações enviadas pelo dispositivo podem ser lidas diretamente no console Application Data da plataforma The Things Network, mas para implementar um broker MQTT e disponibilizar um painel mais legível é possível fazer instalação de um broker MQTT por Node JS e a integração com a plataforma TagoIO suportada pelo The Things Network e disponível de forma gratuita para até 5 dispositivos.

Dentro de sua aplicação no site do TTN clique em Payload Formats. Digite Custom no primeiro campo, selecione decoder e cole o conteúdo do arquivo src/TTN/packed_decoder.js na caixa logo abaixo.

Vá para a aba Integrations no site do TTN e clique em add integration. Selecione "TagoIO" na lista. Preencha o ID como um identificador para a conexão. No campo Chave de acesso, selecione a chave padrão e, em Autorização, insira uma senha individual que será requerida no site do TagoIO.

Faça o login no TagoIO (<https://admin.tago.io/>) ou crie uma nova conta. Após fazer o login e ativar a conta.

Clique em Devices e selecione Adicionar dispositivo. Na seleção a seguir, selecione Custom > The Things Network e atribua o mesmo nome de dispositivo que foi dado no site do TTN/Integrations. No campo Device EUI, copie o valor do mesmo nome no site da TTN (The Things Network) e que já foi usado para configurar o loraconf.h.

2.3 Hardware

A análise de pacotes de requisição de rede por WiFi utiliza a antena WiFi do dispositivo de borda como sensor, inutilizando sua função como transmissor da informação de contagem.

O processamento destas requisições é feito em um processador ESP32 de forma a contabilizar as requisições e descartar as repetições do mesmo dispositivo.

Após a obtenção do número, uma mensagem no formato MQTT é transmitida pela antena LoRa do dispositivo que pode ser visto na Figura 4.

Processadores ESP32 e o protocolo de modulação LoRa consomem pouca energia (consumo de 280mA para taxa de atualização a cada segundo), sendo este dispositivo alimentado por uma bateria de lítio de 1,1Ah de capacidade que permite uma autonomia aproximada de 4hs.

A disposição dos pinos de comunicação da placa utilizada é apresentada na **Figura 5** e mostra que é possível ligar outros sensores no dispositivo de borda em trabalhos futuros.

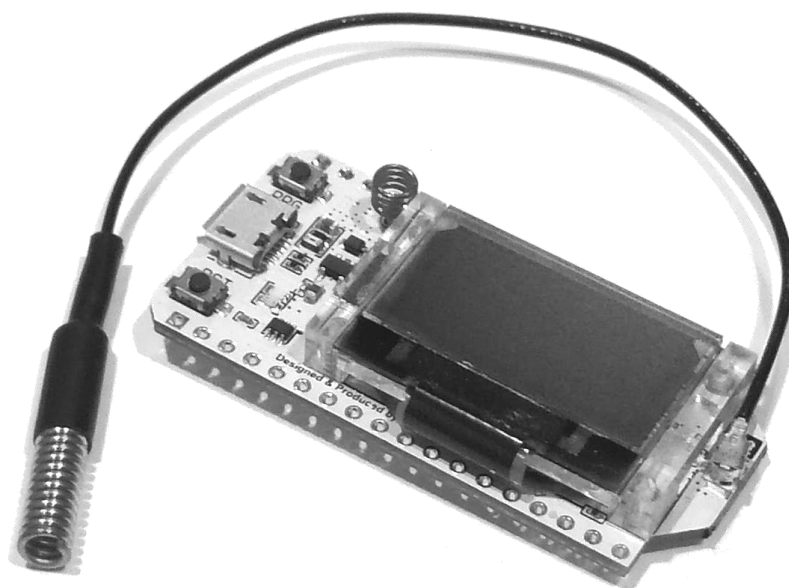


Figura 4 – Dispositivo com antena WiFi, processador ESP32 e antena LoRa.

WIFI LoRa 32(V2) Pinout Diagram

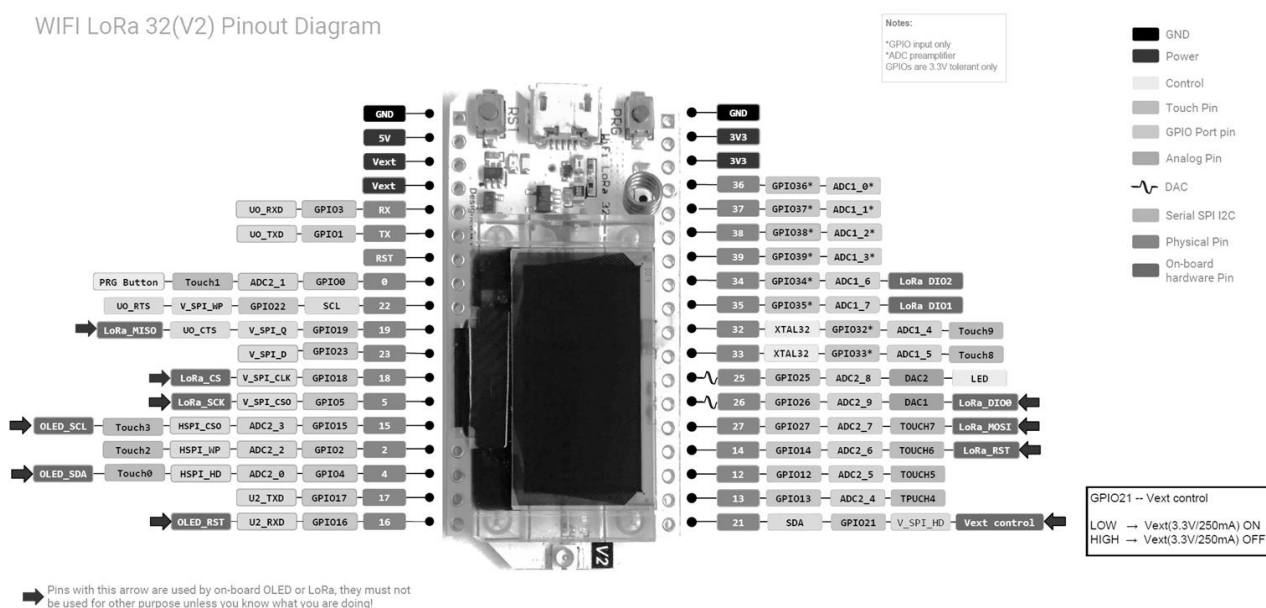


Figura 5 - Diagrama de pinos de comunicação da placa ESP32 utilizada.

2.4 Limites do contador

O contador analisa requisições de conexão WiFi, portanto, caso o dispositivo móvel em posse da pessoa esteja com o recurso WiFi desligado ou se o WiFi estiver com uma conexão estável estabelecida com um roteador, o dispositivo não faz requisição e a pessoa não é contabilizada.

3. RESULTADOS

Testes de alcance de sinal de transmissão LoRa utilizando a antena padrão enviada junto com o produto mantiveram conexão em um raio de 700 metros de distância que equivale a área descrita na Figura 6 no entorno do PESC.



Figura 6 - Área de alcance de comunicação LoRa 433mhz

Este alcance se mostra favorável a aplicações como contador de pessoas que usam os estacionamentos do CT, Letras e Reitoria ou presentes nos pontos de ônibus da Av. Horacio Macedo, CCMN ou Av. Pedro Calmon.

Áreas estas onde os dispositivos móveis estarão procurando redes WiFi, permitindo a contabilização das pessoas e, mesmo distantes, manter a conexão com o Gateway LoRaWAN instalado no PESC.

Apesar de 700 metros ser suficiente para a maioria das aplicações, estudos como os citados por (ALOÿS AUGUSTIN, 2016) com antenas apropriadas mostram um alcance de até 3400 metros.

4. CONCLUSÕES

Este trabalho utilizou as principais características abordadas no desenvolvimento de dispositivos relacionados a Internet das Coisas e demonstra uma gama diversa de aplicações seja como o contador apresentado neste trabalho ou uma evolução com outros sensores instalados nas portas do dispositivo de borda utilizado.

5. REVISÃO BIBLIOGRÁFICA

1 Bibliografia

ALOÿS AUGUSTIN, J. Y. T. C. A. W. M. T. A study of LoRa: Long range & low power networks for the internet of things. **Sensors**, v. 16, n. 9, p. 1466, 2016.

HAOCHAO LI, E. C. L. C. X. G. J. X. K. W. A. L. M. N. Wi-Counter: smartphone-based people counter using crowdsourced Wi-Fi signal data. **IEEE Transactions on Human-Machine Systems**, v. 45, n. 4, p. 442–452, 2015.

THE THINGS NETWORK. Brazil, 2016. Disponível em: <<https://www.thethingsnetwork.org/country/brazil/>>. Acesso em: 2019 Junho 11.

VERKEHRSROT. **ESP32-Paxcounter** | **Hackaday.io**, 2018. Disponível em: <<https://hackaday.io/project/105258-esp32-paxcounter>>. Acesso em: 03 Junho 2019.

6. NOTA DE RESPONSABILIDADE

O autor é o único responsável pelo material impresso neste trabalho.