

Universidade Federal do Amazonas – UFAM
Instituto de Computação
[ICC – 304] Tópicos Especiais em Redes de Computadores
Prof.: Edjair Mota
Relatório - Aula Prática 3

Alunos:

Arlinton José, Joao Alberto, Yuri Assayag , Marcos Rebelo, Mateus Medeiros, Rodrigo Leão

1. Introdução

O trabalho prático 2 consiste na elaboração de uma comunicação entre dispositivos (sensores) utilizando o LoRa. A comunicação se dá através de um dispositivo sensor utilizando arduíno e um dispositivo gateway utilizando um raspberry pi. O nó de envio (sender) é fixado na extremidade do corredor da Faculdade de Tecnologia (FT) enviando 100 pacotes para cada valor de Spreading Factor (SF). O nó de recebimento (Gateway) é inicialmente colocado a 48 metros de distância do sender e realiza a captura dos pacotes (também é preciso ter seu valor de SF configurado), posteriormente variando a distância. A coleta foi feita até o estacionamento do Instituto de Ciências Exatas (ICE) com a distância de 384 metros.

2. Objetivo

O objetivo é analisar os efeitos na potência de sinal recebida, a relação sinal-ruído (SNR), o tempo de comunicação, a taxa de perdas de pacotes, tudo isso variando a distância entre os dispositivos com coletas feitas a cada 48 metros.

Todos esses efeitos são analisados após a mudança de valor do fator de espalhamento (SF). O SF define o uso da técnica de espalhamento na comunicação, assim, grandes sequências de bits são codificadas em um único símbolo, visando diminuir a relação sinal ruído (SNR). Então, aumentando o valor de SF, tende-se a aumentar o SNR, a sensibilidade e o alcance. SF é um valor que pode ser configurado na faixa de 7 a 12.

utilizando a biblioteca RadioHead conforme a especificação do trabalho.

No Setup() as seguintes configurações são necessárias:

- *setTxPower(20)*: Seta a potência de transmissão para 20db.
- *setFrequency(915)*: Seta a frequência para 915MHz
- *setSignalBandwidth(500000)*: Fixa a largura de banda em 500kHz.
- *setSpreadingFactor(X)*: Seta o fator de espalhamento. (No nosso caso esta configuração é alterado a cada 100 pacotes enviados, variando de 7 ao 12).
- *setCodingRate(6)*: Taxa de codificação fixa em 4/5.

Após realizadas as configurações acima, para cada pacote enviado o sender obteve o valor do sensor e o tempo, neste caso através do módulo RTC. O dado enviado pelo sender é no formato: **num-xxx-rtcTime**, sendo *num* o número do pacote, *xxx* o valor enviado pelo sensor e *rtcTime* o tempo atual no formato **hora:minuto:segundo:milésimo**.

Gateway: Responsável por realizar a captura do pacote enviado pelo sender. As mesmas configurações feitas no setup() do sender são necessárias no setup() do receiver. Após obter um pacote, o receiver emite através da porta serial do arduino as informações obtidas. Então, é possível obter a mensagem no formato enviado pelo sender + as informações de RSSI (potência de sinal recebida) e SNR (relação sinal-ruído).

Para fazer a mudança automática do valor de SF, foi necessário enviar um pacote de ordenamento, ou seja, um pacote diferente do que esperávamos receber. Através disso, quando o gateway recebe esse pacote (enviado 20 vezes para não correr o risco de ser perdido), então realiza a mudança do valor de SF e esperava um delay (tempo que o sender precisava para emitir os 20 pacotes de mudança e também mudar o seu valor de SF).

5. Métricas analisadas.

Para cada valor de SF, analisamos:

- Perda de pacote: 100 - quantidade de pacotes recebidos.
- Tempo de recepção: diferença entre o tempo do último pacote recebido pelo primeiro pacote recebido.
- Potência de Sinal: Valor de RSSI na unidade receptora (gateway).
- Relação sinal-ruído: valor de SNR em dBm na unidade receptora (gateway).

6. Análises

Através da análise dos pacotes gerados no arduino, o tamanho do header possui 32 bits e o payload pode variar até 2008 bits dependendo das configurações.

Com a coleta feita a cada distância e variando os valores de SF entre 7 e 14 cada um com 100

pacotes, obtemos os seguintes resultados:

48m				
SF	DIF TEMPO	MÉDIA RSSI	MÉDIA SNR	PERDA
7	21.684	-57.84	5.85	2
8	22.978	-60.60	6.8	2
9	25.792	-59.09	7.0	1
10	30.298	-58.45	6.71	3
11	38.095	-58.38	6.69	1
12	56.025	-58.49	5.91	1

Tabela 2. Distância 1

96m				
SF	DIF TEMPO	MÉDIA RSSI	MÉDIA SNR	PERDA
7	21.925	-67.64	5.81	0
8	23.200	-71.27	6.99	0
9	26.015	-68.25	7.38	0
10	30.591	-68.79	6.42	3
11	38.414	-70.52	6.55	0
12	56.507	-69.68	5.58	0

Tabela 3. Distância 2

144m				
SF	DIF TEMPO	MÉDIA RSSI	MÉDIA SNR	PERDA
7	21.906	-79.74	5.54	0
8	23.198	-84.93	6.73	1
9	26.015	-84.31	7.13	0
10	30.593	-83.93	6.53	2
11	38.415	-81.97	6.14	0
12	56.515	-81.54	5.44	0

Tabela 4. Distância 3

192m				
SF	DIF TEMPO	MÉDIA RSSI	MÉDIA SNR	PERDA
7	21,708	-87,96	5,19	0
8	22,998	-84,01	6,87	0
9	25,783	-82,39	6,9	0
10	30,328	-83,78	6,69	0

11	38,081	-84,53	6,05	0
12	56,015	-82,89	5,18	0

Tabela 5. Distância 4

240m				
SF	DIF TEMPO	MÉDIA RSSI	MÉDIA SNR	PERDA
7	21,465	-94,81	4,3	0
8	23,572	-98,44	4,51	0
9	26,187	-95,94	5,6	0
10	31,109	-94,66	5,38	0
11	39,267	-100,33	3,7	0
12	56,572	-97,15	3,46	0

Tabela 6. Distância 5

288m				
SF	DIF TEMPO	MÉDIA RSSI	MÉDIA SNR	PERDA
7	21,938	-111,43	-2,933	11
8	23,198	-112,99	-3,600	0
9	26,016	-112,69	-3,231	9
10	30,592	-115,33	-5,158	0
11	38,414	-109,87	-1,881	0
12	56,508	-110,81	-2,420	0

Tabela 7. Distância 6

336m				
SF	DIF TEMPO	MÉDIA RSSI	MÉDIA SNR	PERDA
7	21,905	-112,67	-3,22	2
8	23,199	-114,42	-4,22	1
9	26,016	-123,14	-11,73	34
10	30,594	-120,92	-9,60	23
11	38,015	-117,74	-6,78	6
12	55,946	-120,22	-8,52	1

Tabela 8. Distância 7

384m				
SF	DIF TEMPO	MÉDIA RSSI	MÉDIA SNR	PERDA
7	21,915	-115,92	-5,67	40
8	23,196	-118,80	-8,00	14

9	26,015	-121,94	-10,80	17
10	30,621	-118,36	-7,45	0
11	38,448	-118,06	-6,64	0
12	56,51	-119,40	-7,94	0

Tabela 9. Distância 8

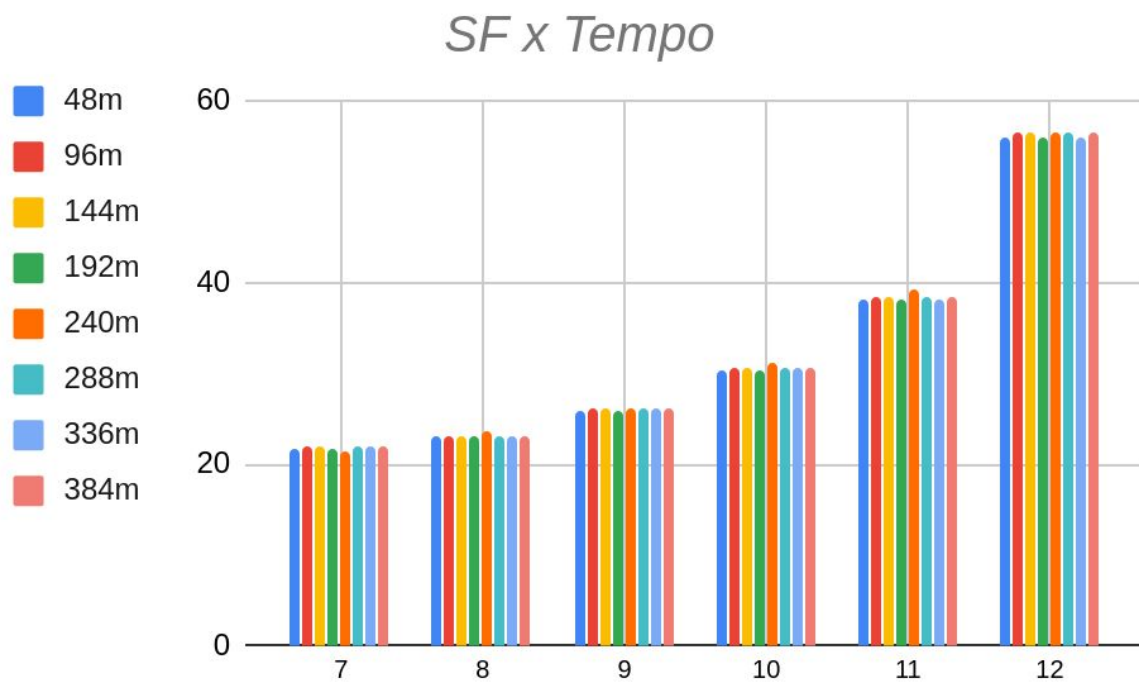


Gráfico 1. Valores de SF pelas diferença de tempo entre pacotes.

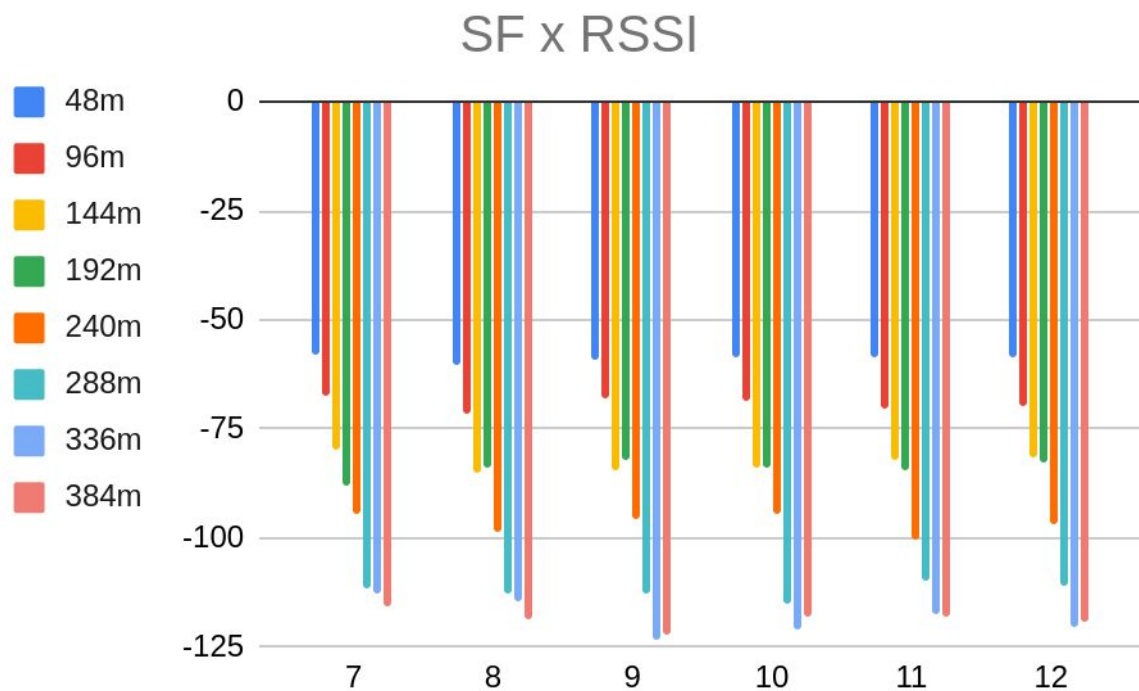


Gráfico 2. Valores de SF pela média de valores de RSSI.

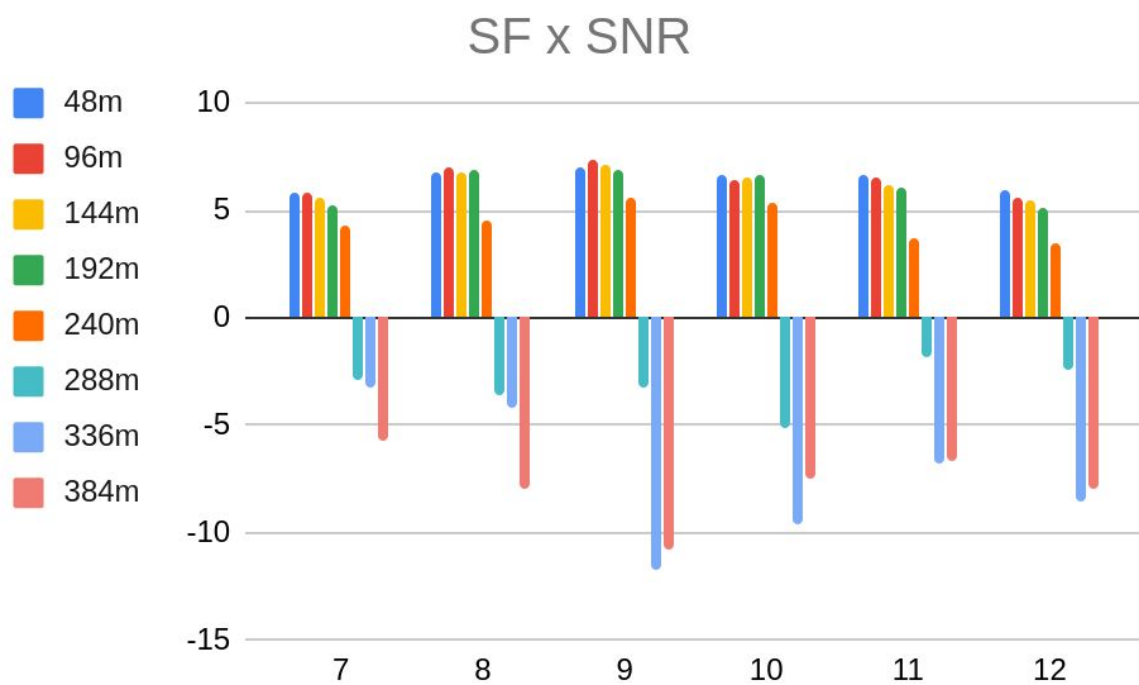


Gráfico 3. Valores de SF pela média dos valores de SNR.

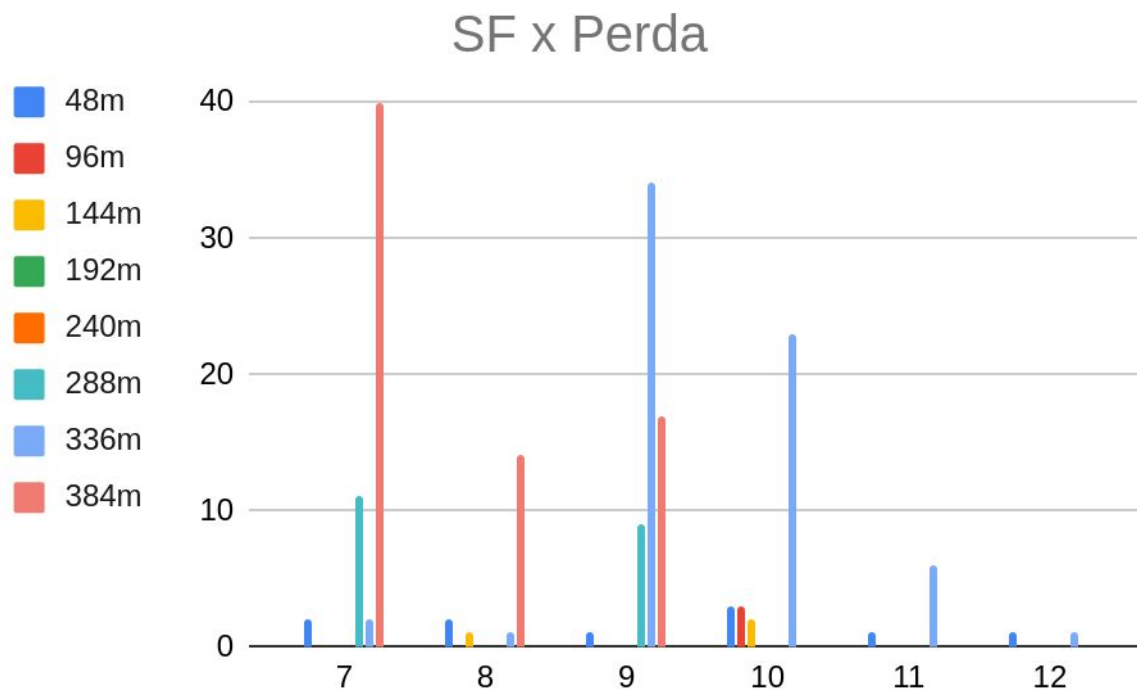


Gráfico 4. Valores de SF pela quantidade de perda de pacotes.

Através dos dados podemos perceber que o melhor valor de SF para curtas distâncias é o 9, pois é o que possui menor perda de pacotes (1 pacote perdido em 48m e nenhum pacote perdido nas distâncias de 96m, 144m e 192m). Como coletamos em distâncias diferentes, então dividimos e as quatro primeiras distâncias são curtas e as 4 últimas são longas. Além dos menores valor de perda de pacotes, o SF 9 é o que possui menor tempo de resposta entre os pacotes e possui os valores de RSSI mais altos.

Já para distâncias mais longas o SF 8 é o melhor valor, pois é o que possui uma quantidade razoável de perdas (14 perdas para a maior distância), embora não seja o SF que possui menor perda já que o SF 11 e 12 possuem menores. Porém, levando em consideração o tempo entre os pacotes (23ms contra 56ms do SF 12 e 38ms do SF 11 por exemplo) e o valor de SNR (-5, contra -6 e -7 do SF 11 e 12 respectivamente), então analisando todos os parâmetros, o SF 8 é que obtém vantagem.

Em relação a influência do SF no tempo de recepção, podemos observar através do gráfico 1 que aumentando o valor de SF aumenta-se também o tempo de recepção dos pacotes. Isso se dá devido a importância do SF na comunicação, já que aumentando o SF aumenta-se também o tempo de recepção pois o tamanho do pacote é afetado e isso possibilita uma menor perda já que os pacotes ficam mais resistentes a ruídos e serão lidos de maneira mais confiável, ao custo de menor taxa de dados e mais congestionamento.

6.0 Conclusão

Conforme o experimento realizado podemos observar o comportamento do LoRa de acordo com a distância entre os dispositivos usados. Com isso, podemos analisar a influência de determinados parâmetros na comunicação, em especial o fator de espalhamento (SF), que impacta tanto no valor de RSSI da transmissão, quanto o tamanho do pacote, a quantidade de perdas e a relação sinal-ruído.

Esses fatos são importantes quando é necessário configurar a comunicação para determinados locais, levando em consideração a confiabilidade ou a velocidade da transmissão.

7.0 Referências

[1] (2019). Como usar com arduino módulo real time clock - rtc. <https://blogmasterwalkershop.com.br/arduino/como-usar-com-arduino-modulo-real-time-clock-rtc-ds3231/>. Accessed: 2019-11-19.

[2] (2019). Iotlabs: Exploring lora technology. <http://tet.pub.ro/pages/altele/Docs/Shield%20Dragino%20Lora/Lora%20Shield%20-%20Wiki%20for%20Dragino%20Project.pdf>. Accessed: 2019-11-19

[3] (2019a). Lora com raspberry e arduino. [lora-com-raspberry-e-arduino/](https://www.dobitaobyte.com.br/lora-com-raspberry-e-arduino/). Accessed: 2019-11-19. <https://www.dobitaobyte.com.br/>

[4] (2019b). Lora shield and rpi to build a lorawan gateway. <https://www.instructables.com/id/Use-Lora-Shield-and-RPi-to-Build-a-LoRaWAN-Gateway/>. Accessed: 2019-11-19.

[5] (2019c). Lora tester for raspberry - a useful repository. <https://github.com/lupyuen/LoRaArduino>. Accessed: 2019-11-19.

[6] (2019d). Manual do lora shield. http://wiki.dragino.com/index.php?title=Lora_Shield. Accessed: 2019-11-19.

[7] (2019). pyradiohead github page. <https://github.com/exmorse/pyRadioHeadRF95>. Accessed: 2019- 11-19.

[8] (2019). Radiohead main page. <https://www.airspayce.com/mikem/arduino/RadioHead/index.html>. Accessed: 2019-11-19.

[10] Embarcados (2019). Lora shield dra gino+ raspberry. <https://www.embarcados.com.br/lora-arduino-raspberry-pi-shield-dragino/>. Accessed: 2019-11-19.

8.0 Anexos



Imagem 1 - Coleta no estacionamento - Sender com Sensor



Imagem 2 - Coleta em pontos intermediários

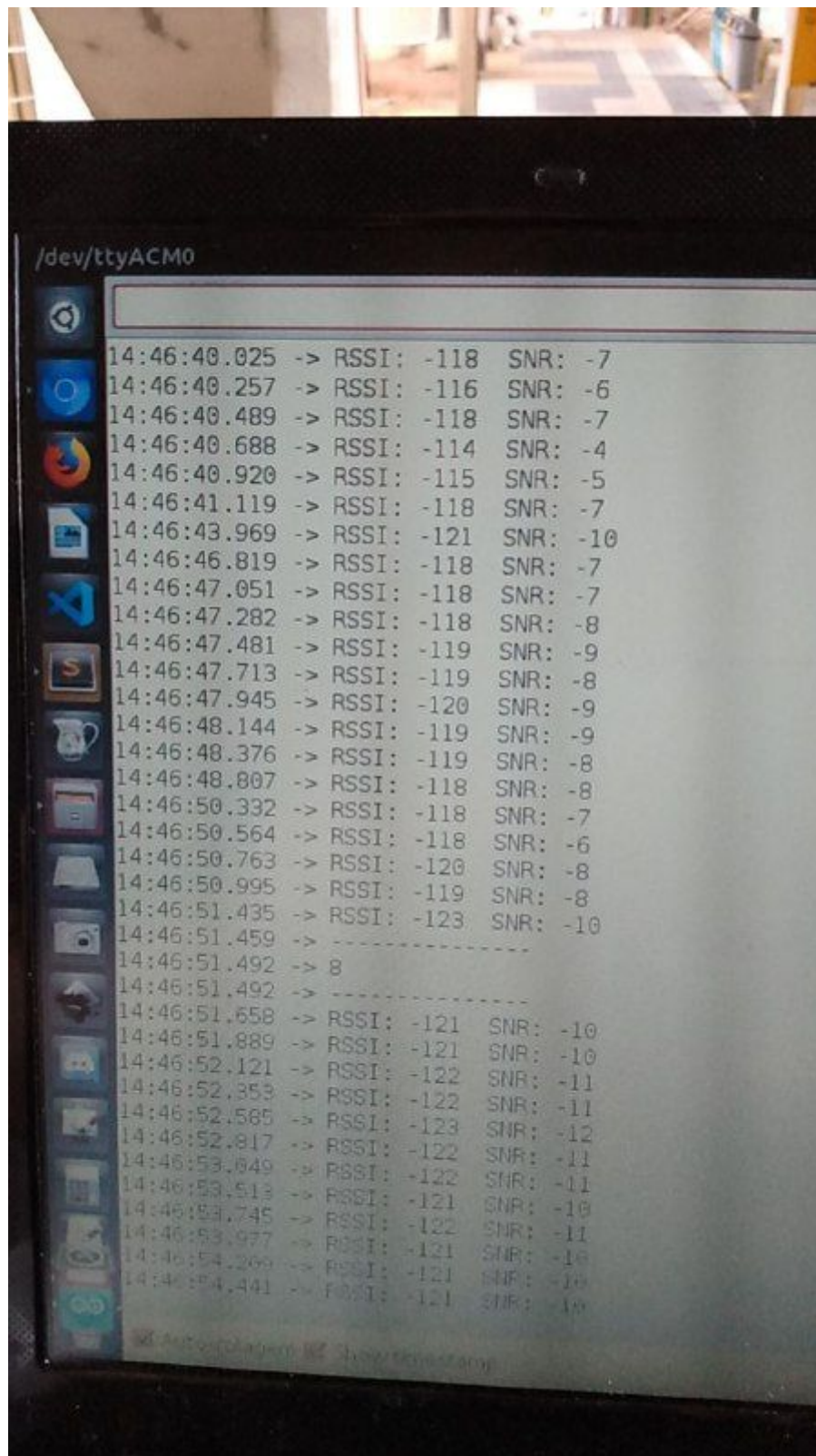


Imagem 3 - Coleta no Receptor - Lora

