

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ELETRÔNICA E COMPUTAÇÃO
CURSO DE ENGENHARIA DE COMPUTAÇÃO**

Felipe Jaskulski Maia

**RECEPTOR DE FREQUÊNCIA SIMPLES DE BAIXO CUSTO COM
AQUISIÇÃO E PROCESSAMENTO DE DADOS BRUTOS DE
CONSTELAÇÕES GPS**

Santa Maria, RS
2018

Felipe Jaskulski Maia

**RECEPTOR DE FREQUÊNCIA SIMPLES DE BAIXO CUSTO COM AQUISIÇÃO E
PROCESSAMENTO DE DADOS BRUTOS DE CONSTELAÇÕES GPS**

Trabalho de conclusão de curso apresentado ao Curso de Engenharia de Computação, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do título de **Bacharel em Engenharia de Computação**.

Orientador: Prof. Dr. José Eduardo Baggio

Santa Maria, RS
2018

Felipe Jaskulski Maia

**RECEPTOR DE FREQUÊNCIA SIMPLES DE BAIXO CUSTO COM AQUISIÇÃO E
PROCESSAMENTO DE DADOS BRUTOS DE CONSTELAÇÕES GPS**

Trabalho de conclusão de curso apresentado ao Curso de Engenharia de Computação, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do título de **Bacharel em Engenharia de Computação**.

Aprovado em 04 de maio de 2018:

José Eduardo Baggio, Dr. (UFSM)
(Presidente/Orientador)

Carlos Henrique Barriquello, Dr. (UFSM)

Cesar Augusto Prior, Dr. (UFSM)

Santa Maria, RS
2018

AGRADECIMENTOS

Primeiramente gostaria de agradecer aos meus pais, que me deram o suporte necessário para concluir o curso por toda a sua duração. Gostaria também de agradecer aos meus amigos mais próximos que me ajudaram nos momentos mais difíceis e comemoraram juntos nos momentos de maior alegria de minha graduação. Por fim, e não menos importante, gostaria de agradecer aos professores e à universidade por compartilhar de seus conhecimentos e dos meios de obtê-los ao longo de minha jornada para que isso me tornasse um profissional competente na área da engenharia.

RESUMO

RECEPTOR DE FREQUÊNCIA SIMPLES DE BAIXO CUSTO COM AQUISIÇÃO E PROCESSAMENTO DE DADOS BRUTOS DE CONSTELAÇÕES GPS

AUTOR: Felipe Jaskulski Maia
ORIENTADOR: Prof. Dr. José Eduardo Baggio

Esse trabalho apresenta um sistema de baixo custo e portátil, que adquire mensagens de constelações de satélites GPS e gera um arquivo com dados brutos destas mensagens. Considera-se que o receptor trabalha somente com frequência simples (L1), e os dados são obtidos com o uso de uma antena que opera na frequência de 1.575,42 MHz, com amplificador de baixo ruído alimentado por uma fonte de 5 V. O sinal obtido pela antena é direcionado a um módulo RTL2832U v3 que transmite as mensagens dos satélites para o computador no qual esse módulo está conectado.

Esse computador processa as mensagens do módulo utilizando o software livre GNSS-SDR, e fornece os arquivos no formato RINEX com os dados brutos. Esses arquivos podem ser utilizados como informação de entrada em um aplicativo que adquira o posicionamento do sistema no momento em que foi feita a aquisição dos dados, como o RTKLib, fazendo uso de técnicas de pós-processamento.

Palavras-chave: GPS. Obtenção de Dados Brutos. Frequência Simples. RINEX. Posicionamento. RTL2832U v3. GNSS-SDR. Pós-processamento.

ABSTRACT

LOW-COST SINGLE FREQUENCY RECEIVER WITH ACQUISITION AND PROCESSING OF RAW DATA FROM GPS CONSTELLATIONS

AUTHOR: Felipe Jaskulski Maia
ADVISOR: Prof. Dr. José Eduardo Baggio

This work presents a portable low-cost system that acquires messages from GPS constellations' satellites and generates a raw data file from these messages. Considering that the receiver works in single frequency (L1), the data is acquired using an antenna that operates at 1,574.42 MHz, with a low noise amplifier fed by a 5 V power supply. The signal acquired by the antenna is directed to a RTL2832U v3 dongle that transmits the satellite messages to a computer on which this dongle is connected.

This computer processes the messages from the dongle utilizing the GNSS-SDR open source software and provides RINEX files with the raw data. These files can be utilized as input on an application that provides the system position in the moment the data was acquired, such as RTKLib, making use of post-process techniques.

Keywords: GPS. Raw data. Single frequency. RINEX. Positioning. RTL2832U v3. GNSS-SDR. Post-process.

SUMÁRIO

| | |
|--|----|
| AGRADECIMENTOS | 4 |
| RESUMO | 5 |
| ABSTRACT | 6 |
| SUMÁRIO | 7 |
| 1 INTRODUÇÃO | 8 |
| 2 JUSTIFICATIVA E REFERENCIAL TEÓRICO | 11 |
| 3 METODOLOGIA | 19 |
| 4 RESULTADOS | 24 |
| 5 DISCUSSÃO | 29 |
| 6 REFERÊNCIAS | 30 |
| APÊNDICE A: INSTALAÇÃO E CONFIGURAÇÃO DO SOFTWARE GNSS-SDR . | 32 |
| APÊNDICE B: CONFIGURAÇÃO E UTILIZAÇÃO DOS APLICATIVOS RTKPLOT E RTKPOST | 34 |

1 INTRODUÇÃO

Desde a época das grandes navegações percebe-se a importância do posicionamento global para a humanidade. Desde lá, muitos avanços ocorreram nessa área, melhorando cada vez mais a precisão do posicionamento no globo. Hoje, um avanço considerado significativo na estimativa da posição significa uma melhoria na ordem de metros ou até centímetros, e isso se deve ao crescente aumento ao acesso à informação que dispomos.

Hoje, fazemos uso de Sistemas de Navegação Global por Satélite (GNSS) que utilizam as mensagens enviadas por satélites para obter a localização de um receptor para a maioria das necessidades de posicionamento. Esses sistemas possuem erros em sua comunicação com os receptores. Entre esses, destacam-se os erros por propagação pela Troposfera e pela Ionosfera. O Instituto Nacional de Pesquisas Espaciais – Centro Regional Sul (INPE-CRS) elaborou um simulador de concentração eletrônica e de conteúdo eletrônico acumulado baseado no modelo SUPIM (Sheffield University Plasmasphere-Ionosphere Model).

SUPIM é um modelo físico que descreve a distribuição de ionização que ocorre ao longo do tempo na ionosfera terrestre, em altitudes de aproximadamente 130 Km ao longo de linhas fechadas de campo magnético [...]. O modelo fornece como saída as densidades, velocidades alinhadas ao campo magnético e temperaturas para os elétrons e para os íons O⁺, H⁺, He⁺, N₂⁺, O₂⁺ e NO⁺. Esses dados são fornecidos em função da latitude e altura para um intervalo de tempo determinado, a partir de uma data e hora pré-estabelecidas. (PETRY, A., 2010, p. 6)

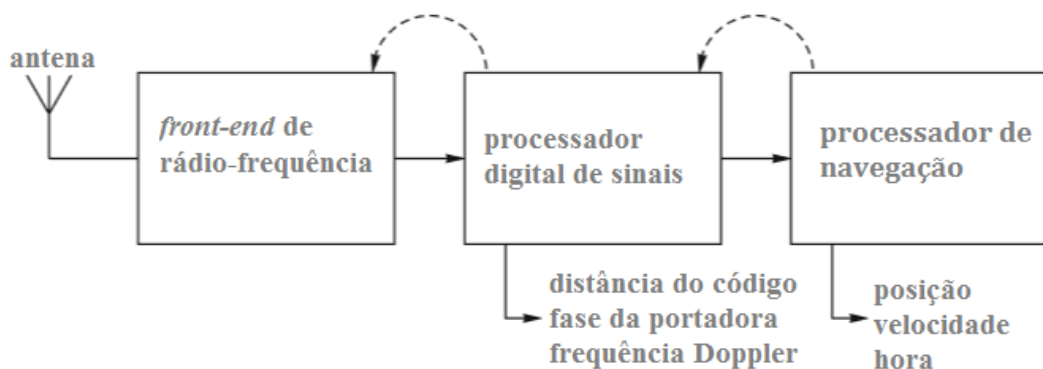
Esse simulador necessita ser alimentado por arquivos de dados brutos, e um receptor que forneça esse tipo de arquivo não é encontrado no mercado. Por isso o intuito desse projeto é fornecer os arquivos para que, não só o simulador, mas mais pesquisas possam ser implementadas com o uso desses arquivos.

Tendo isso em vista, esse trabalho objetiva, além de fornecer os dados brutos transmitido pelos satélites, obter um receptor de frequência simples de baixo custo e portátil. Para isso, também deve-se estudar e entender como funcionam os Sistemas Globais de Navegação por Satélite (GNSS) e um Receptor Definido por Software

(SDR). A partir disso, os próximos passos são encontrar componentes de baixo custo que sejam suficientes para oferecer os dados necessitados pelo SDR a partir de uma antena que receba dados do Sistema Global de Posicionamento (GPS) – sistema esse que está incluso nos GNSS. Feito isso, deve-se buscar programas que façam a interface com a antena e o dispositivo a ser conectado no computador (GNSS-SDR e RTL-SDR e suas respectivas dependências). Por fim, deve-se executar o programa por tempo suficiente para adquirir os dados necessários para se obter um arquivo no formato RINEX com os dados brutos para que as técnicas de pós-processamento de um software que calcule o posicionamento com esses dados possam ser analisadas ou então para que novas técnicas que utilizem esses dados brutos possam ser empregadas.

Buscando a obtenção de um arquivo no formato RINEX, que é o formato utilizado internacionalmente para troca de informações de dados brutos de sistemas de navegação por satélite, o receptor a ser elaborado nesse projeto necessita de suporte de hardware (uma antena, um *front-end* de radiofrequência e um processador digital de sinais) e software (processamento de navegação), bem como a compreensão da configuração e do funcionamento desses, conectados conforme ilustrado na Figura 1. Além disso, se faz necessário o conhecimento de como são dispostas as mensagens enviadas pelos satélites na banda L1 (1.575,42 MHz) – uma das bandas de frequência em que os satélites enviam mensagens – para que a aquisição dos dados seja suficiente para o fornecimento do arquivo de dados brutos.

Figura 1 - Diagrama de Blocos de Um Receptor GNSS



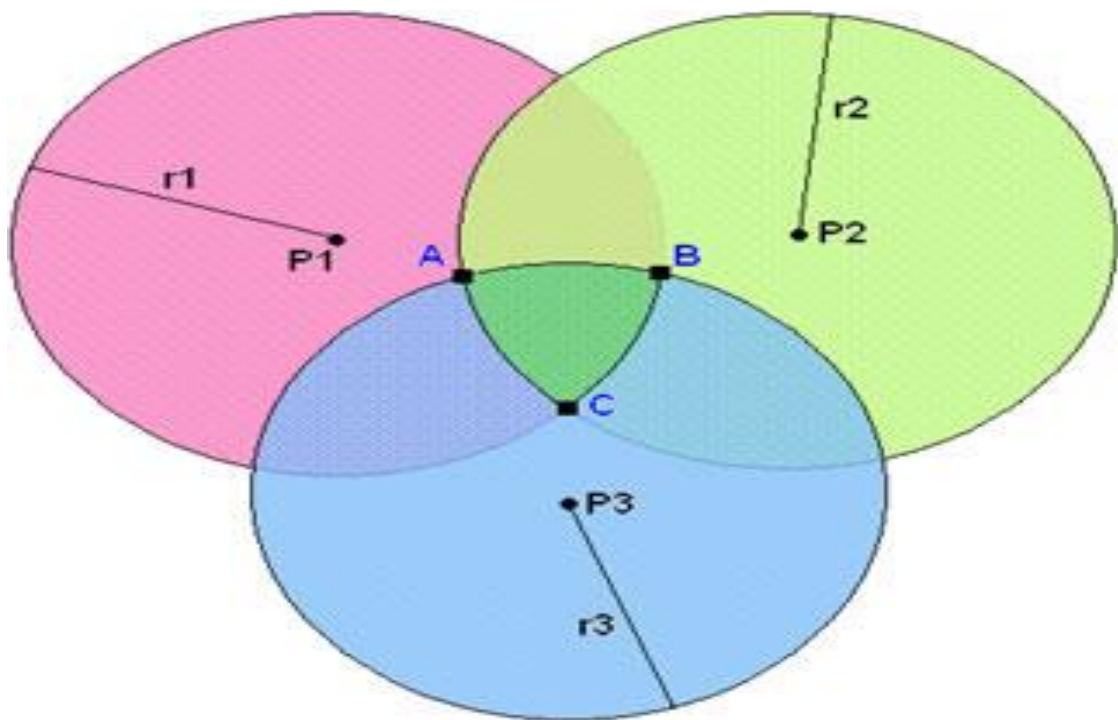
Fonte: Hofmann-Wellenhof, Lichtenegger, Wasle (2008), tradução nossa.

Por fim, deve ser feito um estudo de como essas informações podem ser exibidas e demonstrar a utilização dos dados brutos.

2 JUSTIFICATIVA E REFERENCIAL TEÓRICO

O padrão mais popular dentre os GNSS (Sistema Global de Navegação por Satélite) é o GPS (Sistema Global de Posicionamento), que é o padrão americano. Além do GPS existem outros padrões, como o GLONASS (padrão russo), Galileo (padrão europeu), BeiDou (padrão chinês), QZSS (padrão japonês), entre outros. Por ser o mais conhecido, o padrão abordado nesse projeto foi o GPS. Ele faz uso do método de trilateração para obter o posicionamento do receptor. Esse método consiste em determinar a distância entre pelo menos três satélites e o receptor, como é mostrado na figura abaixo:

Figura 2 - Demonstração do Método de Trilateração



Fonte: Teleco: Inteligência em Telecomunicações¹

Vale evidenciar que a posição do receptor está localizada na região da intersecção das estimativas das distâncias entre os 3 satélites e o receptor. O cálculo dessa distância é dado pelo intervalo de tempo na propagação do sinal enviado pelo satélite até o receptor. Portanto, são necessários pelo menos observações de quatro

¹ Disponível em: http://www.teleco.com.br/tutoriais/tutoriallogsm2/pagina_2.asp. Acesso em mai. de 2018.

satélites, três para o cálculo do posicionamento pela trilateração e um para a sincronização da hora no relógio interno do receptor com a hora dos relógios internos dos satélites.

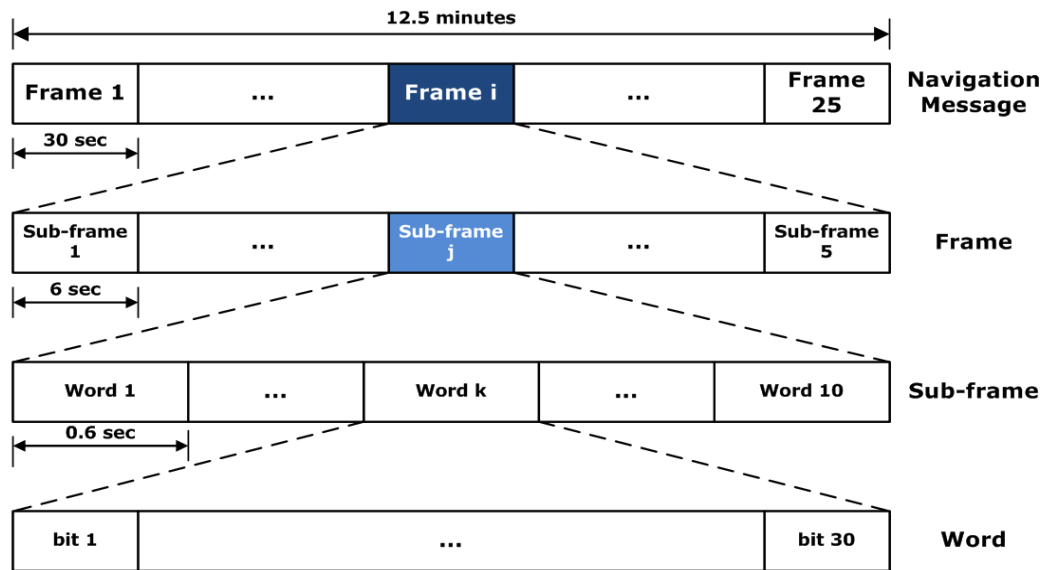
Esse método está sujeito a algumas fontes de erro. Dentre elas, as principais são:

- Desvios nas camadas de Ionosfera e Troposfera – os sinais eletromagnéticos dos satélites passam por desvios de trajetória e velocidade causadas por íons, na Ionosfera, e por diferença de densidade do ar, na Troposfera;
- Diluição de precisão (DOP) – um fenômeno causado pela proximidade dos satélites que realizarão a comunicação com o receptor. Essa proximidade pode aumentar a faixa de intersecção da distância entre os satélites e o receptor, o que prejudica a precisão do posicionamento;
- Multi-caminho – erro causado por sinais refletidos provocados por objetos que se encontram perto da antena e que interferem com o sinal do satélite;
- Erros nos relógios dos satélites – apesar dos satélites possuírem relógios atômicos, que são relógios de precisão elevada, eles ainda ocasionam erro. Considerando que o sinal eletromagnético viaja na velocidade da luz – aproximadamente 300.000 km/s – um nanossegundo de erro no relógio resulta em um erro de cerca de 30 cm na medição da distância para um satélite;
- Erro nas efemérides – as efemérides são as informações necessárias para se obter a localização de um satélite. Apesar de os satélites estarem em órbitas muito precisas, o insuficiente conhecimento do campo gravitacional terrestre, das forças gravitacionais do Sol e da Lua e do atrito remanescente da atmosfera terrestre, bem como a pressão das radiações solares nos satélites provoca variações em suas órbitas, provocam esse erro;
- Ruído do receptor – um receptor está limitado à capacidade de seu próprio hardware. Isso faz com que o sinal por ele recebido esteja sujeito a ruídos do próprio receptor, o que ocasiona erros no sinal.

Os satélites de constelações GPS transmitem constantemente mensagens em 5 bandas de frequências distintas: L1 (1.575,42 MHz), L2 (1.227,60 MHz), L3 (1.381,05 MHz), L4 (1.379,913 MHz) e L5 (1.176,45 MHz). Dentre essas, somente as bandas L1 e L2 são de uso civil. São as bandas que os receptores de frequência simples e dupla utilizam para obter informações a fim de definir seus posicionamentos.

A mensagem da banda L1 dos satélites GPS é composta por 25 *frames*, que são subdivididos em 5 *subframes* compostos de 10 palavras de 30 bits cada. A taxa de transmissão de bits é de 50 bps, ou seja, cada palavra leva 0,6 segundos para ser enviada, um *subframe* leva 6 segundos, um *frame* 30 segundos, e, por fim, uma mensagem leva 12,5 minutos. No primeiro *subframe* são enviados três coeficientes para um polinômio quadrático que modela a correção do clock do satélite, o estado de vida do satélite e tempo de vida do dado, uma estimativa do atraso do grupo de sinais, o número de semanas GPS – contagem de semanas desde 6 de janeiro de 1980 – e uma predição da precisão do intervalo de distâncias entre o usuário e o satélite. No segundo e no terceiro são enviados os dados de efemérides – informações referentes a precisão e estado de vida do satélite, coeficientes de correção do clock do satélite, parâmetros orbitais e o número de semanas GPS, com validade de 2 horas antes e depois do dado ser computado pelo segmento de controle do GNSS, e usado para computação de coordenadas do satélite em tempo real – do satélite. O quarto *subframe* provê os parâmetros do modelo ionosférico necessário para os cálculos de refração ionosférica dos receptores de frequência simples, as informações UTC (Tempo Coordenado Universal), parte do *almanac* – informações referentes ao posicionamento do satélite, porém com menor precisão orbital que as efemérides, com validade de até 90 dias e usado para acelerar o momento da primeira correção (tempo necessário para um receptor GPS adquirir os sinais do satélite e os dados de navegação, e calcular uma solução de posição) em 15 segundos – e informações reservadas para uso militar. O último *subframe* contém o restante do *almanac* e o estado de até outros 24 satélites da constelação. Os três primeiros *subframes* são retransmitidos em cada *frame*. Já os *frames* 4 e 5, que contém as diferentes páginas que compõe a mensagem de navegação, são alterados a cada *frame* e possuem uma taxa de repetição de 25 *frames*. Todos os satélites enviam a mesma informação nos *subframes* 4 e 5. Então para uma mensagem completa ser transmitida, é necessário um tempo de 12,5 minutos ou 750 segundos. Esse tempo é exibido pelo software durante a execução.

Figura 3 - Composição da Mensagem de Navegação



Fonte: Página Navipedia².

As mensagens enviadas através da banda L2 não são necessárias para a compreensão desse trabalho, pois o receptor projetado, por ser de frequência simples, trabalha somente com a banda L1.

Receptores de frequência simples, são mais baratos. O receptor eTrex 10, da Garmin, possui preço sugerido pela própria fabricante de R\$ 699,00. Em compensação por seu preço menor, ele possui uma precisão menor (1 a 3 metros) por demorar mais para resolver as ambiguidades de posicionamento. Já receptores de frequência dupla, como a estação eletrônica GTS-100N, da Topcon, são capazes de obter precisão de aproximadamente 2 mm + 2 ppm, porque são capazes de receber mensagens de duas frequências (L1 e L2) e isso faz com que as ambiguidades do posicionamento sejam resolvidas mais rapidamente. Porém, uma estação eletrônica como essa é encontrada em sites de compra e venda, como o Mercado Livre, por cerca de R\$ 13.500,00³, que muitas vezes pode ser considerado um valor alto a se pagar por uma precisão maior. Por isso, é raro o uso de receptores de frequência

² Disponível em: http://www.navipedia.net/images/f/fe/Navigation_Message.png; Acesso em dez. 2017.

³ Disponível em: https://produto.mercadolivre.com.br/MLB-702655561-estaco-total-topcon-gts-102n-melhor-que-trimble-gps-rtk-_JM. Acesso em mai. de 2018.

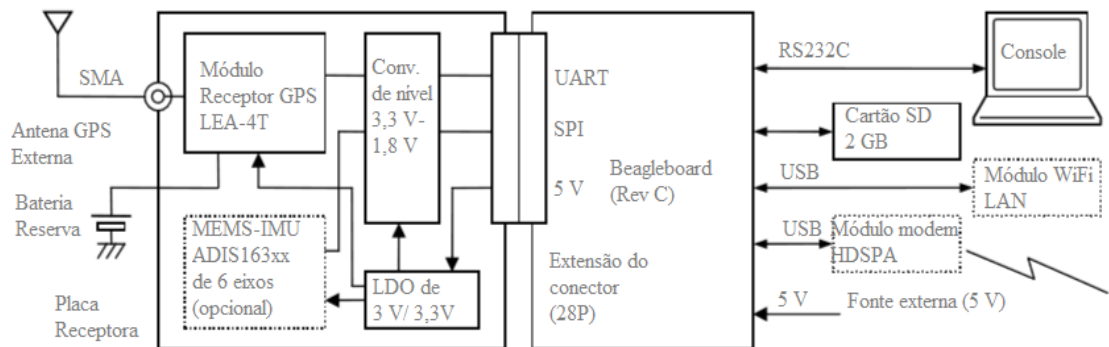
dupla em pesquisas ou até em funções em que a precisão de um receptor de frequência simples seja suficiente, portanto não foi utilizado nesse trabalho.

Um sistema de posicionamento GPS padrão encontrado no mercado, como o eTrex 10, fornece somente informações referentes ao posicionamento do aparelho, o que difere do receptor proposto nesse projeto. O maior diferencial do receptor proposto é o fato de ele fornecer dados brutos das mensagens enviadas pelos satélites, sendo que dados nessa forma podem ser úteis para pesquisas com novas técnicas de correção de posicionamento ou até para o estudo do funcionamento das técnicas empregadas atualmente. Além disso esse receptor também possui portabilidade para outros sistemas e baixo custo quando comparado ao receptor de frequência simples evidenciado anteriormente.

Para executar o projeto, temos que ter em mente que “[...] um receptor GNSS é composto por 3 blocos funcionais: o *front-end* de radiofrequência; o processador digital de sinais; e o processador de navegação” (HOFFMANN-WELLENHOF; LICHTENEGGER; WASLE, 2008, p. 85).

Com um projeto similar, Takasu e Yasuda (Development of the low-cost RTK-GPS receiver with an open source program package RTKLIB, 2009) demonstram que é possível executar o projeto de um receptor GPS de frequência simples de baixo custo e que forneça dados brutos. Para o *front-end* de seu projeto, eles utilizam o receptor LEA-4T disponibilizado pela u-blox AG por possuir as seguintes características: boa performance, ser um módulo pequeno e compacto e de baixo consumo de potência, prover a resolução da ambiguidade da fase da portadora em meio ciclo e suportar uma taxa de atualização dos dados brutos de até 10 Hz. Essa taxa de atualização é importante para projetos de receptores RTK – Cinemático em Tempo Real – pois, como o próprio nome diz, a aquisição e o processamento dos dados deve ser feita em tempo real.

Figura 4 - Receptor RTK-GPS: Esquemático



Fonte: Takasu, T.; Yasuda A. (2009), tradução nossa.

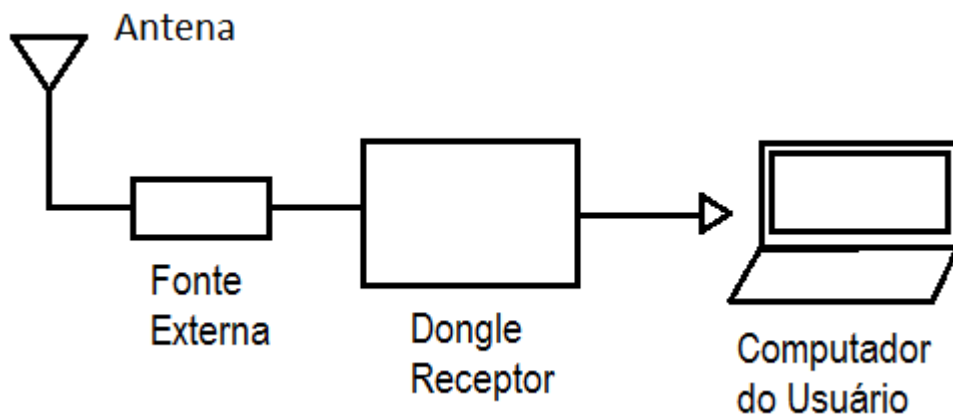
Além desse receptor, o projeto citado também utiliza de um BeagleBoard – computador de placa única – que executa Linux embarcado para executar a biblioteca de aplicativos RTKLIB com interface de linha de comando a fim de prover comunicação com um console através de uma porta RS232C, com um cartão SD ou com um módulo WiFi ou HSDPA via USB, e prover a alimentação de 5 V requisitada pelo sistema.

Figura 5 - Receptor RTK-GPS: *BeagleBoard* e Receptor

Fonte: Takasu, T., Yasuda A. (2009).

Como o foco do projeto proposto nesse trabalho é prover os arquivos de dados brutos para pós-processamento, foi possível utilizar um receptor ainda mais barato do que o receptor do projeto citado. Além disso, não é necessário o computador de placa única, pois a comunicação feita no projeto utiliza somente a saída USB do módulo receptor e a porta USB do computador do usuário. Porém, a fonte de tensão que alimenta o amplificador da antena deve ser acoplada entre o módulo receptor e a antena e ocupar mais uma porta USB do computador utilizado para prover a alimentação necessária, e essa conexão entre a fonte, a antena e o módulo pode ser dificultada pelas conexões dessa fonte e ser necessário encontrar um adaptador para conectar o módulo a essa fonte. Com isso resolvido, precisamos de uma referência para o processador de navegação e exemplos de módulos mais baratos e simples que os citados no projeto anterior.

Figura 6 - Esquemático Geral do Projeto



Durante a *IPNTJ Summer School on GNSS*, Suzuki (GNSS-SDRLIB: Introduction and Practice of GNSS Software Receiver, 2014) demonstra como deve ser feita a configuração de softwares de receptores de GNSS para diversas constelações e também para constelações específicas separadamente, para que possa ser executado o processamento de navegação. Ele mostra que, apesar do consumo de potência ser maior e de algumas funções dependerem do processador MIPS utilizado, um software receptor possui algumas vantagens importantes sobre um hardware receptor, como re-programabilidade, re-configurabilidade e menor custo

de hardware. Além disso, todos softwares exibidos por ele nessa *summer school* são código aberto, ou seja, não incrementam em nada o custo do sistema.

Nessa *summer school* também são exibidos alguns dos principais hardwares utilizados como *front-end* para sistemas como o feito nesse trabalho. De acordo com Suzuki, os principais aspectos que devem ser observados na escolha de um *front-end* adequado são:

- Preço;
- Quantidade de bandas (ou quantidade de *front-ends*);
- Taxa de amostragem;
- Largura de banda do sinal;
- Alcance de frequência (somente L1 ou não);
- Interface de conexão (USB 2.0, USB 3.0, Ethernet...);
- Bits de amostragem;
- Precisão do oscilador.

Dentre os principais *front-ends*, é destacado o RTL2832U como sendo o aparelho de SDR (radio definido por software) mais famoso por possuir uma grande comunidade e apresentar baixo custo. Também é destacado sobre esse aparelho o fato de ele possuir uma baixa precisão de *clock* e de não poder utilizar uma antena ativa por padrão, portanto deve-se realizar uma alteração no hardware para que ele possa alimentar o amplificador de baixo ruído da antena ou então adicionar uma fonte de tensão para alimentar esse amplificador.

A seguir, é feito um detalhamento sobre como o sistema foi montado, como foi feita a escolha dos componentes, qual o módulo *front-end* escolhido, como tornar o sistema operacional e como obter os resultados finais.

3 METODOLOGIA

O primeiro passo para esse projeto é o estudo prévio dos componentes que devem ser utilizados. Após uma breve pesquisa, foi constatado que um kit com placa de desenvolvimento – conjunto de ferramentas embarcadas em uma placa controladas por um microcontrolador e que visam desenvolver um sistema embarcado para solucionar um problema específico – poderia nos fornecer o posicionamento, porém, não forneceria os dados brutos. Isso não nos seria útil, pois um dos objetivos do projeto é fornecer os dados brutos das mensagens dos satélites para que exista a possibilidade de visualizar como são empregadas as técnicas de correção de erro na mensagem enviada pelo satélite até o receptor. Desta forma, buscou-se sistemas embarcados para esse uso específico, e algumas opções surgiram com preços variando entre 26 e 1.100 dólares, porém não foram encontradas nenhuma dessas opções no Brasil. Como outro dos objetivos do projeto é desenvolver um receptor de baixo custo, escolhemos o embarcado de menor preço (RTL2832U v3), pois ele opera com sinais em uma faixa de frequências que contém a banda L1 (1.575,42 MHz) e possui um *clock* mais preciso que suas versões anteriores. Em contrapartida possui uma frequência de amostragem menor, se comparado com outros embarcados mais caros, o que acaba prejudicando na precisão do posicionamento final.

Outro componente importante a ser adquirido é a antena GPS. Ela deve operar na banda L1, possuir um amplificador de baixo ruído (LNA) e possuir um conector compatível com o do sistema embarcado usado. A antena foi possível de ser encontrada no Brasil custando 25 reais, porém ela necessita de uma fonte de tensão de 5 V para operar, já que o sistema embarcado não fornece a tensão de saída necessária para alimentar o LNA da antena. Alguns desses componentes foram escolhidos a partir de uma *summer school* de T., Suzuki (2014), que leciona como desenvolver um receptor GNSS de frequência simples.

Figura 7 - Hardware do Receptor GPS



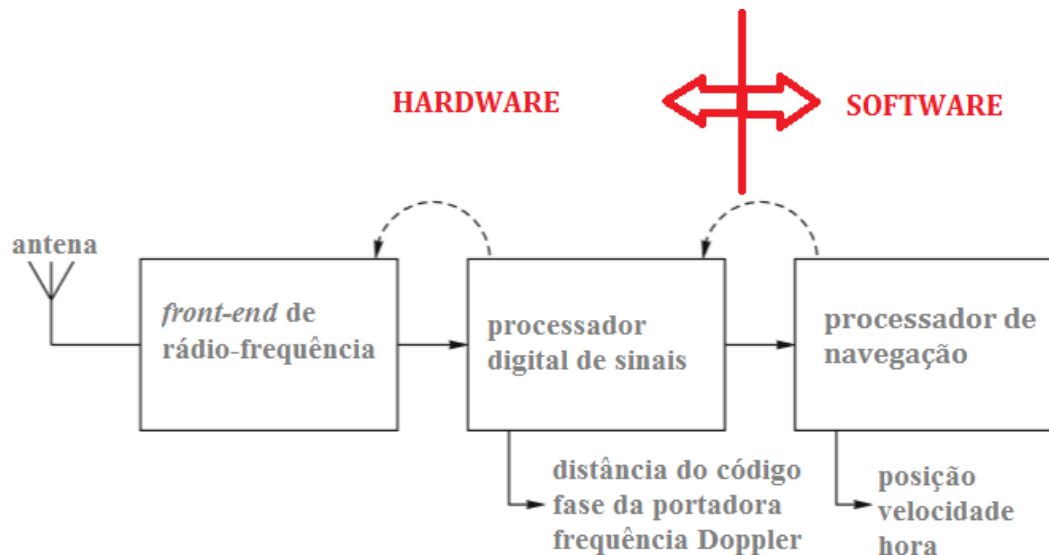
Alguns fatores importantes diferenciam o projeto lecionado por Suzuki (GNSS-SDRLIB: Introduction and Practice of GNSS Software Receiver, 2014), em sua *summer school*, do projeto executado nesse trabalho. Um deles é o *front-end* escolhido, que apesar de ser semelhante, o RTL2832U v3 possui uma precisão de *clock* melhorada em comparação com a sua primeira versão ao custo de ser um pouco mais caro. Também é importante ressaltar que esse módulo receptor possui um processador digital de sinais que possibilita que os dados de navegação do satélite sejam transmitidos para o computador que processará esses dados de navegação. Além disso, a nova versão também possibilita habilitar seu *bias T* – componente de três portas utilizado para prover tensão ou corrente contínua em circuitos de radiofrequência –, usado para alimentar o LNA da antena, via software, enquanto a primeira versão não permitia isso como padrão, só podendo habilitar removendo um diodo do hardware e soldando um indutor de 47 nH em um local especificado em uma imagem por Suzuki. Porém, devido a perdas causadas pelo comprimento do condutor da antena utilizada, foi necessário alimentar seu LNA com um componente externo, a fonte de tensão já mencionada anteriormente.

O software escolhido foi o GNSS-SDR por possuir algumas qualidades que seriam de grande vantagem para o projeto: é código aberto desenvolvido em C++;

apresenta portabilidade; possibilita executar tarefas tanto em tempo real quanto pós-processadas; apresenta boa performance e como medi-la; leva em consideração a arquitetura específica de cada processador; e pode fazer uso de processadores com múltiplos núcleos.

Dessa forma, os três blocos que compõem um receptor GNSS foram estabelecidos: o *front-end* de radiofrequência e o processador digital de sinais por hardware, e o processador de navegação por software.

Figura 8 - Decomposição do Projeto em Hardware e Software



Fonte: Hofmann-Wellenhof, Lichtenegger, Wasle (2008).

Outra diferença no projeto é a escolha do sistema operacional utilizado para executar o SDR (Receptor Definido por Software). Enquanto Suzuki leciona como executar o SDR em uma versão do Windows, nesse projeto foi preferido fazer o uso de um sistema baseado em Linux por ser um sistema de mais fácil acesso, portanto, de maior portabilidade. Mais precisamente, o sistema operacional utilizado nesse projeto foi o Ubuntu 16.04 LTS, que é uma versão do Ubuntu que permite a instalação do software escolhido mais facilmente. O procedimento de instalação do software pode ser observado na seção Apêndice A, nas páginas finais desse trabalho.

Para verificar se os dados recebidos são coerentes, é preciso saber como estão dispostos os dados nos arquivos RINEX. Os arquivos de saída para um receptor de

frequência simples com dados de uma constelação GPS são dois: o arquivo de dados de observação (extensão “.17O”) e o arquivo de dados de navegação (extensão “.17N”). Ambos consistem de duas seções: a seção do cabeçalho e a seção dos dados. O cabeçalho contém *labels* nas colunas 61 até 80 para cada linha da seção do cabeçalho. Algumas informações podem ser conferidas já no cabeçalho, como constelação, data e hora da primeira observação. Após a linha com o *label* “END OF HEADER”, começa a seção dos dados, que é onde estão as informações utilizadas pelas técnicas de correção de posicionamento, como pode ser visto posteriormente nas figuras 9 e 10.

A seção de dados do arquivo de dados de observação provê informações referentes ao tipo de observação. Sua primeira linha informa, entre espaços, o ano, mês, dia, hora, minutos, segundos, uma flag de eventos e o número de satélites que enviaram os dados, respectivamente. Depois disso, cada linha informa os dados enviados por cada satélite, até que a mensagem do último satélite seja enviada. Se existirem 6 satélites, 6 linhas contendo informações de cada satélite é disposta. A primeira informação é qual satélite enviou (ex: G02). Depois disso, são mostrados os dados de cada tipo de observação. Esses tipos são listados na seção de cabeçalho do arquivo. Esses dados são disponibilizados de forma a serem lidos por declarações do formato FORTRAN.

Já a seção de dados do arquivo de dados de navegação, é organizada de uma forma diferente. Em sua primeira linha, há a identificação do satélite e momento em que os dados foram adquiridos (ano, mês, dia, hora, minutos e segundos). A seguir, iniciam-se 29 números que indicam dados referentes ao satélite, na hora indicada pelo início da mensagem. Ao fim desses 29 números, são demonstradas informações sobre o próximo satélite ou sobre o próximo horário de aquisição de dados de navegação. São esses 29 números que indicam as informações de cada *subframe* da mensagem do satélite detalhados no capítulo anterior.

Para fazer a análise dos resultados, a biblioteca de aplicativos RTKLib foi utilizada. Com os aplicativos RTKPost – usado para obtenção de posicionamento utilizando técnicas de pós-processamento – e RTKPlot – aplicativo que exhibe os resultados de posicionamento – foi possível obter um arquivo de texto exibindo as coordenadas de posições obtidas pelo software que utiliza os dados brutos como entrada e também um mapa de pontos, exibindo de maneira visual como foram distribuídos esses pontos. Primeiramente deve-se configurar o RTKPost inserindo

nele como arquivos de entrada os arquivos no formato RINEX obtidos pelo receptor. Essas configurações podem ser observadas no Apêndice B, encontrado nas páginas finais desse trabalho.

Os resultados obtidos com a metodologia empregada podem ser observados na seção seguinte.

4 RESULTADOS

As figuras 9 e 10 abaixo representam as informações contidas nos arquivos RINEX. Percebe-se à direita das imagens o que cada uma das linhas do cabeçalho representa, e que após a linha que informa o fim do cabeçalho imediatamente iniciam as informações utilizadas pelos métodos de correção de posicionamento.

Figura 9 - Arquivo de Observação

```

3.02      OBSERVATION DATA      G      RINEX VERSION / TYPE
G = GPS  R = GLONASS  E = GALILEO  S = GEO  M = MIXED  COMMENT
GNSS-SDR      UNKNOWN USER      20171129 151245 UTC  PGM / RUN BY / DATE
GPS OBSERVATION DATA FILE GENERATED BY GNSS-SDR  COMMENT
GNSS-SDR VERSION 0.0.9      COMMENT
See http://gnss-sdr.org      COMMENT
DEFAULT MARKER NAME      MARKER NAME
UNKNOWN USER      CTTC      OBSERVER / AGENCY
GNSS-SDR      Software Receiver  0.0.9      REC # / TYPE / VERS
Antenna number      Antenna type      ANT # / TYPE
0.0000      0.0000      0.0000      APPROX POSITION XYZ
0.0000      0.0000      0.0000      ANTENNA: DELTA H/E/N
G      4 C1C L1C D1C S1C      SYS / # / OBS TYPES
DBHZ      SIGNAL STRENGTH UNIT
2017  11  29  17  11  26.1448020  GPS      TIME OF FIRST OBS
18  18  137  7      LEAP SECONDS
END OF HEADER

> 2017 11 29 17 11 26.1448020 0 6
G02 20389042.735 6 5005.268 6 -334.829 6 41.974
G06 22991722.781 6 34059.792 6 -2286.785 6 39.972
G126085073518.952 7 -16225.384 7 1099.638 7 45.393
G24 20895997.362 7 24317.735 7 -1654.693 7 44.923
G25 22405287.940 6 -39858.171 6 2652.768 6 38.745
G296940916731.992 6 -29535.765 6 2034.677 6 41.412
> 2017 11 29 17 11 26.2448020 0 6
G02 20389059.490 6 5038.727 6 -334.593 6 41.411
  
```


Figura 10 - Arquivo de Navegação

```

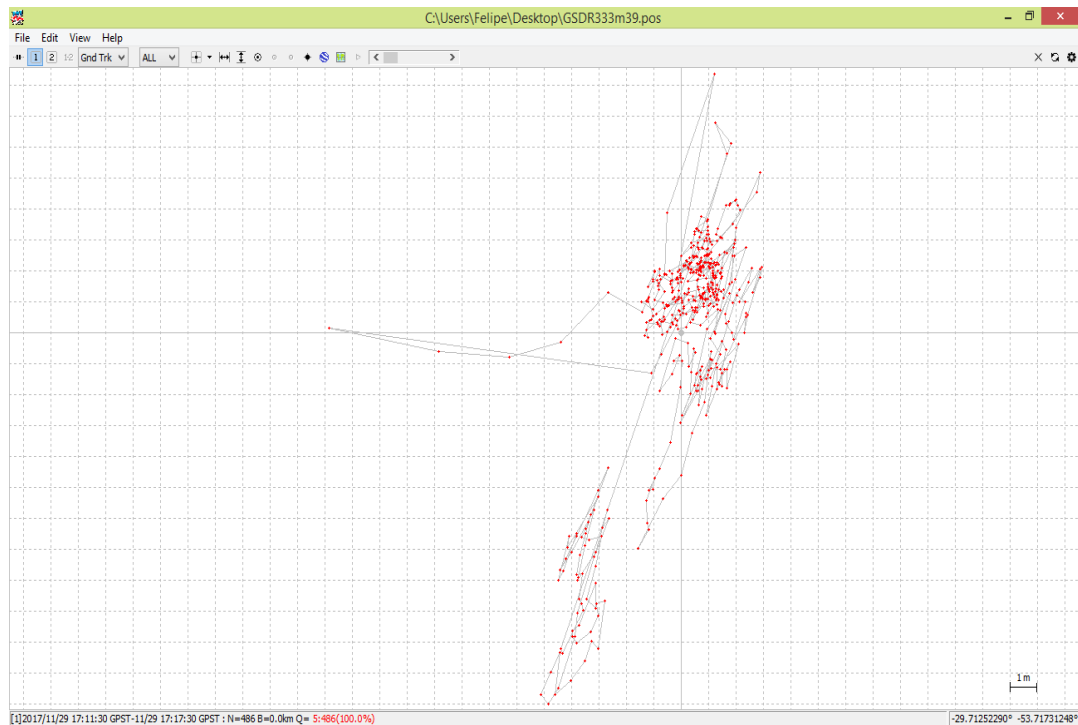
3.02          N: GNSS NAV DATA      G: GPS          RINEX VERSION / TYPE
GNSS-SDR      UNKNOWN USER          20171129 151245 UTC PGM / RUN BY / DATE
GPS NAVIGATION MESSAGE FILE GENERATED BY GNSS-SDR      COMMENT
GNSS-SDR VERSION 0.0.9                                COMMENT
See http://gnss-sdr.org                               COMMENT
GPSA          .1024D-07  -.1490D-07  -.5960D-07   .1192D-06      IONOSPHERIC CORR
GPSB          .9830D+05  -.1475D+06  -.1966D+06   .8520D+06      IONOSPHERIC CORR
GPUT         -.1862645149D-08  .000000000D+00  503808 1209   TIME SYSTEM CORR
              18      18      137      7              LEAP SECONDS
                                                    END OF HEADER

G02 2017 11 29 18 00 00  .268280040473D-03  -.989075488178D-11  .00000000000D+00
      .98000000000D+02  .26562500000D+02  .525879047830D-08  .152718699633D-02
      .135600566864D-05  .173540810356D-01  .603497028351D-05  .515362396049D+04
      .32400000000D+06  -.223517417908D-06  .450884560717D+00  .189989805222D-06
      .948438447891D+00  .26103125000D+03  -.193371243717D+01  -.878215152615D-08
      .436446751188D-09  .10000000000D+01  .19770000000D+04  .10000000000D+01
      .00000000000D+00  .00000000000D+00  -.204890966415D-07  .98000000000D+02
      .31987800000D+06  .40000000000D+01

G06 2017 11 29 18 00 00  .411683693528D-03  .125055521494D-11  .00000000000D+00
      .30000000000D+01  .20906250000D+02  .469162399640D-08  -.137745839884D+00
      .908970832825D-06  .682740821503D-03  .569038093090D-05  .515378744888D+04
      .32400000000D+06  -.279396772385D-07  .499874503882D+00  -.242143869400D-07
      .969150573720D+00  .27237500000D+03  -.128650066997D+01  -.824962934446D-08
      .411445709794D-09  .10000000000D+01  .19770000000D+04  .10000000000D+01
      .00000000000D+00  .00000000000D+00  .465661287308D-08  .30000000000D+01
      .31975800000D+06  .40000000000D+01
  
```

Com as informações contidas desses arquivos, utilizam-se técnicas de correção de posicionamento para que sejam obtidas as coordenadas de latitude e longitude referentes à localização do receptor no momento da captura das informações. Um mapa de pontos com essas coordenadas pode ser visto na figura 11.

Figura 11 - Mapa de Pontos do RTKPlot



Esse mapa de pontos foi obtido utilizando o arquivo de posicionamento resultante do pós-processamento dos dados brutos com correção de erros ionosféricos e troposféricos pré-definidos no RTKPlot. Como podemos ver na figura 12, esse arquivo nos informa qual a latitude, longitude e altitude de cada um dos pontos da figura 11. Com essa informação é possível obter quantitativamente a posição do ponto médio, bem como a média da distância entre cada um dos pontos e o ponto médio. Para isso, primeiramente foi calculada a média aritmética de todas as latitudes, longitudes e altitudes. Depois, uma conversão dessa média e também de cada valor individual de latitude, longitude e altitude para valores Euclidianos (em metros) com origem no centro da Terra foi necessária para que se fizesse o cálculo da distância entre cada um dos pontos do mapa da figura 11 e o ponto médio calculado. A conversão das coordenadas para valores Euclidianos é dada pelas seguintes fórmulas:

$$X = \cos(\text{lon}) * \cos(\text{lat}) * (RT^5 - \text{alt})$$

$$Y = \text{sen}(\text{lat}) * (RT - \text{alt})$$

$$Z = \text{sen}(\text{lon}) * \text{sen}(\text{lat}) * (RT - \text{alt})$$

Esse cálculo de distância foi baseado na fórmula de Pitágoras para calcular a distância entre dois pontos em um plano real tridimensional.

$$D = \sqrt{X^2 + Y^2 + Z^2}$$

Em seguida, a média aritmética dessas distâncias foi adquirida, para que assim pudéssemos demonstrar a precisão da correção ionosférica, e assim provar que o método para adquirir os arquivos de dados brutos opera satisfatoriamente.

Figura 12 - Arquivo de Posicionamento RTKPost

```

% program : RTKPOST ver.2.4.3 b28
% inp file : C:\Users\Felipe\Desktop\GSDR333m39.170
% inp file : C:\Users\Felipe\Desktop\GSDR333m39.17N
% obs start : 2017/11/29 17:11:26.1 GPST (week1977 321086.1s)
% obs end : 2017/11/29 17:17:30.0 GPST (week1977 321450.0s)
% pos mode : single
% elev mask : 15.0 deg
% lonos opt : broadcast
% tropo opt : saastamoinen
% ephemeris : broadcast
% navl sys : gps glonass
%
% (lat/lon/height=WGS84/ellipsoidal,Q=1:fix,2:float,3:sbas,4:dgps,5:single,6:ppp,ns=# of satellites)
% GPST latitude(deg) longitude(deg) height(m) Q ns sdn(m) sde(m) sdu(m) sdne(m) sdeu(m) sdu(m) age(s) ratio
2017/11/29 17:11:30.145 -29.712510290 -53.717296938 100.5487 5 6 4.4900 3.9116 13.3532 2.4736 -4.1811 -5.1665 0.00 0.0
2017/11/29 17:11:30.245 -29.712512230 -53.717301769 101.0937 5 6 4.4899 3.9116 13.3532 2.4736 -4.1810 -5.1663 0.00 0.0
2017/11/29 17:11:30.345 -29.712511991 -53.717299299 101.5458 5 6 4.4899 3.9116 13.3531 2.4735 -4.1808 -5.1661 0.00 0.0
2017/11/29 17:11:30.445 -29.712512628 -53.717300334 101.3857 5 6 4.4898 3.9115 13.3530 2.4735 -4.1807 -5.1660 0.00 0.0
2017/11/29 17:11:30.545 -29.712510444 -53.717298772 100.7246 5 6 4.4897 3.9115 13.3529 2.4734 -4.1805 -5.1658 0.00 0.0
2017/11/29 17:11:30.645 -29.712513341 -53.717299490 101.4470 5 6 4.4897 3.9115 13.3528 2.4733 -4.1803 -5.1656 0.00 0.0
2017/11/29 17:11:30.745 -29.712511265 -53.717301018 99.9128 5 6 4.4896 3.9115 13.3527 2.4733 -4.1802 -5.1655 0.00 0.0
2017/11/29 17:11:30.845 -29.712509721 -53.717301589 99.6338 5 6 4.4895 3.9115 13.3527 2.4732 -4.1800 -5.1653 0.00 0.0
2017/11/29 17:11:30.945 -29.712512417 -53.717305089 98.4553 5 6 4.4895 3.9115 13.3526 2.4732 -4.1798 -5.1651 0.00 0.0
2017/11/29 17:11:31.045 -29.712514314 -53.717304437 98.4169 5 6 4.4894 3.9114 13.3525 2.4731 -4.1797 -5.1650 0.00 0.0
2017/11/29 17:11:31.145 -29.712512096 -53.717305911 98.0681 5 6 4.4894 3.9114 13.3524 2.4730 -4.1795 -5.1648 0.00 0.0
2017/11/29 17:11:31.245 -29.712513231 -53.717310118 97.9321 5 6 4.4893 3.9114 13.3523 2.4730 -4.1794 -5.1646 0.00 0.0
2017/11/29 17:11:31.345 -29.712515555 -53.717308610 98.1206 5 6 4.4892 3.9114 13.3522 2.4729 -4.1792 -5.1644 0.00 0.0
2017/11/29 17:11:31.445 -29.712519247 -53.717314257 98.8001 5 6 4.4892 3.9114 13.3522 2.4728 -4.1790 -5.1643 0.00 0.0
2017/11/29 17:11:31.545 -29.712521932 -53.717314155 98.8958 5 6 4.4891 3.9113 13.3521 2.4728 -4.1789 -5.1641 0.00 0.0
2017/11/29 17:11:31.645 -29.712521256 -53.717311009 99.0539 5 6 4.4890 3.9113 13.3520 2.4727 -4.1787 -5.1639 0.00 0.0
2017/11/29 17:11:31.745 -29.712516393 -53.717308297 98.4720 5 6 4.4890 3.9113 13.3519 2.4727 -4.1786 -5.1638 0.00 0.0
2017/11/29 17:11:31.845 -29.712515419 -53.717307506 98.8345 5 6 4.4889 3.9113 13.3518 2.4726 -4.1784 -5.1636 0.00 0.0
2017/11/29 17:11:31.945 -29.712516329 -53.717307507 98.5329 5 6 4.4888 3.9113 13.3517 2.4725 -4.1782 -5.1634 0.00 0.0

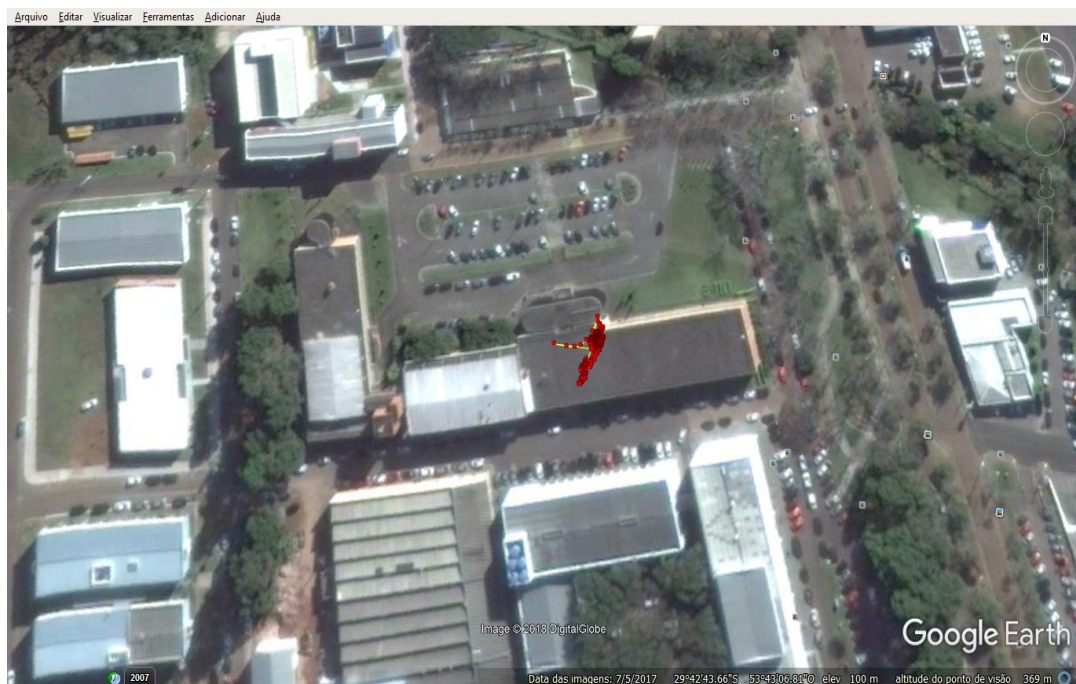
```

⁴ Latitude e Longitude dados em graus. Altitude dada em metros.

⁵ RT – raio da terra em metros. Aproximadamente 6.371.000 metros.

Com os cálculos para verificar o ponto médio, obtivemos uma latitude média de $-29,71252299339508^\circ$, longitude média de $-53,71731272995471^\circ$ e altitude média de 103,2409430041153 m. As coordenadas Euclidianas obtidas para esse ponto médio com origem no centro da Terra são (3.274.528,240608692; -3.157.827,835987347; -4.460.554,300361567) metros. Por fim, a distância média entre os pontos e o ponto médio encontrada foi 3,485506141723842 metros. Como podemos observar, o valor de distância média se encontra próximo do valor de precisão do receptor GPS eTrex 10, o que demonstra que o método para obtenção de dados brutos é satisfatório. Na figura 12 também podemos observar a disposição dos pontos em uma imagem de satélite, de forma a se analisar o posicionamento por um referencial visual empírico.

Figura 13 - Mapa de Pontos Inserido em Imagem de Satélite



Fonte: Google Earth.

A seguir, uma breve discussão acerca dos resultados obtidos.

5 DISCUSSÃO

Podemos verificar que os resultados de posicionamento obtidos são similares aos resultados de um aparelho de localização GPS no mercado. Porém, o objetivo do trabalho não era obter a melhor localização, e sim um projeto de baixo custo e portátil que fornecesse os dados brutos das mensagens enviadas pelos satélites. O custo total desse projeto foi de aproximadamente duzentos e cinquenta reais, o que é um valor baixo se comparado com o receptor GPS indicado nesse trabalho. Com softwares *open source* e de código aberto, o custo não foi incrementado além do valor gasto com hardware, e as técnicas empregadas para correção do posicionamento podem ser analisadas com as informações obtidas dos arquivos no formato RINEX, o arquivo de posicionamento fornecido pelo RTKPlot e uma análise dos códigos da biblioteca de aplicativos RTKLib, o que mostra que o receptor projetado nesse trabalho fornece os dados desejados, cumprindo seu objetivo.

6 REFERÊNCIAS

GNSS-SDR – An Open Source GNSS Software Defined Receiver. Disponível em: <gnss-sdr.org>. Acesso em: 12 de dez. de 2017.

GURTNER, W.; ESTEY, L.; **RINEX – The Receiver Independent Exchange Format Version 3.00.** Disponível em <<http://epic.awi.de/29985/1/Gur2007a.pdf>>. Acesso em: 12 de dez. de 2017.

HOFFMAN-WELLENHOF, B.; WASLE, E.; LICHTENEGGER, H. **GNSS – Global Navigation Satellite Systems: GPS, GLONASS, Galileo and More.** Austria: SpringerWienNewYork, 2008.

LEICK, A.; RAPOPORT, L.; TATARNIKOV, D. **GPS Satellite Surveying.** Ed. 4ª; Hoboken, NJ. John Wiley & Sons, Inc., 2015.

RTL-SDR. Disponível em: <rtl-sdr.com>. Acesso em: 12 de dez. de 2017.

SUZUKI, T. **GNSS-SDRLIB: Introduction and Practice of GNSS Software Receiver.** IPNTJ Summer School on GNSS, 2014. Tokyo Univ. of Marine Science and Technology.

Takasu, T. **RTKLIB ver. 2.4.2 Manual.** Disponível em: <http://www.rtklib.com/prog/manual_2.4.2.pdf>. Acesso em: 12 de dez. de 2017.

Takasu, T.; Yasuda A. **Development of the low-cost RTK-GPS receiver with an open source program package RTKLIB.** International Symposium on GPS/GNSS, International Convention Center, Jeju, Korea, 2009.

Garmin. Disponível em: < <https://www.garmin.com/en-US/> >. Acesso em: 13 de mai. de 2018.

PETRY, A. **ANÁLISE E PARALELIZAÇÃO DO CÓDIGO SUPIM.** São José dos Campos: INPE, 2010.

PETRY, A., DE SOUZA, J., DE CAMPOS VELHO, H. **SISTEMA PARA PREVISÃO OPERACIONAL DA DINÂMICA DA IONOSFERA BASEADO NO MODELO SUPIM.** São José dos Campos: INPE, 2011.

ESTAÇÃO TOTAL ELETRÔNICA: TOPCON Série GTS-110N. Disponível em : < <https://www.embratop.com.br/wp-content/uploads/2017/07/GTS-102-em-Portugu%C3%AAs.pdf> >. Acesso em 13 de mai. de 2018.

TELECO: INTELIGÊNCIA EM TELECOMUNICAÇÕES. Disponível em: < <http://www.teleco.com.br/> >. Acesso em 13 de mai. de 2018.

REVISTA DA GESTÃO COSTEIRA INTEGRADA. Disponível em: < <http://www.aprh.pt/rgci/> >. Acesso em 13 de mai. de 2018.

NOVATEL. Disponível em: < <https://www.novatel.com/#latestNews> >. Acesso em 13 de mai. de 2018.

WERNER, G., ESTEY, L. **RINEX: The Receiver Independent Exchange Format Version 3.01.** Disponível em < <ftp://igs.org/pub/data/format/rinex301.pdf> >. Acesso em 13 de mai. de 2018.

COLLEGE OF ENGINEERING & APPLIED SCIENCE: UNIVERSITY OF COLORADO BOULDER. Disponível em < <https://www.colorado.edu/engineering/> >. Acesso em 13 de mai. de 2018

APÊNDICE A: INSTALAÇÃO E CONFIGURAÇÃO DO SOFTWARE GNSS-SDR

Analisando a documentação e a aplicação que seria empregada junto com esse software, os seguintes passos foram tomados para sua instalação em um computador com o sistema operacional Ubuntu 16.04 LTS:

- Inserir a linha “deb-src <http://us.archive.ubuntu.com/ubuntu/> xenial universe” ao arquivo “/etc/apt/sources.list”;
- Executar os comandos “\$ sudo apt-get update” e “\$ sudo apt-get build-dep gnss-sdr” em uma janela do terminal;
- Adquirir os arquivos de instalação da biblioteca *OsmoSDR* manualmente executando o comando “\$ git clone [git://git.osmocom.org/osmo-sdr.git](https://git.osmocom.org/osmo-sdr.git)” no terminal;
- Acessar o diretório “osmo-sdr/software/libosmosdr” e criar a pasta “build”;
- Acessar a nova pasta “build” via terminal e executar os comandos “\$ cmake ../”, “\$ make”, “\$ sudo make install” e “\$ sudo ldconfig” para instalar a biblioteca *OsmoSDR*;
- Adquirir os arquivos de instalação do bloco fonte do *GNU Radio* manualmente executando o comando “\$ git clone [git://git.osmocom.org/gr-osmosdr](https://git.osmocom.org/gr-osmosdr)” no terminal;
- Acessar o diretório “gr-osmosdr” e criar a pasta build;
- Acessar a nova pasta “build” via terminal e executar os comandos “\$ cmake ./ -Wno-dev”, “\$ make”, “\$ sudo make install” e “\$ sudo ldconfig” para instalar o bloco fonte do *GNU Radio*;
- Instalar as dependências *orc-0.4*, *python 2.7*, *python six* e *Doxygen*;
- Adquirir os arquivos de instalação do software GNSS-SDR manualmente executando o comando “\$ git clone <https://github.com/gnss-sdr/gnss-sdr>”;
- Acessar o diretório “gnss-sdr/build” via terminal e executar os comandos “\$ cmake -DENABLE_OSMOSDR=ON ../”, “\$ make” e “\$ sudo make install” para instalar o software GNSS-SDR;
- Executar os comandos “sudo volk_profile” e “sudo volk_gnssdr_profile” para que o software leve em conta a arquitetura do processador;

- Adquirir os arquivos de configuração do RTL2832U manualmente executando “\$ git clone https://github.com/osmocom/rtl-sdr” no terminal;
- Acessar o diretório “rtl-sdr” e criar a pasta “build”;
- Acessar a nova pasta “build” e executar os comandos “\$ cmake ./”, “\$ make”, “\$ sudo make install” e “\$ sudo ldconfig” para instalar os arquivos de configuração do RTL2832U;
- Criar um arquivo de extensão “.conf” com as configurações do receptor de acordo com a documentação de configuração do GNSS-SDR⁶;
- Abrir uma nova janela de terminal e executar o comando “\$ rtl_tcp -a 127.0.0.1 -f 1575420000 -g 0 -s 2000000”;
- Na primeira janela de terminal, acessar a pasta onde se encontra o arquivo de extensão “.conf” criado e executar o comando “\$ gnss-sdr – config_file=./<nome do arquivo>”, substituindo o termo “<nome do arquivo>” pelo nome do arquivo de extensão “.conf” criado (incluindo a extensão).

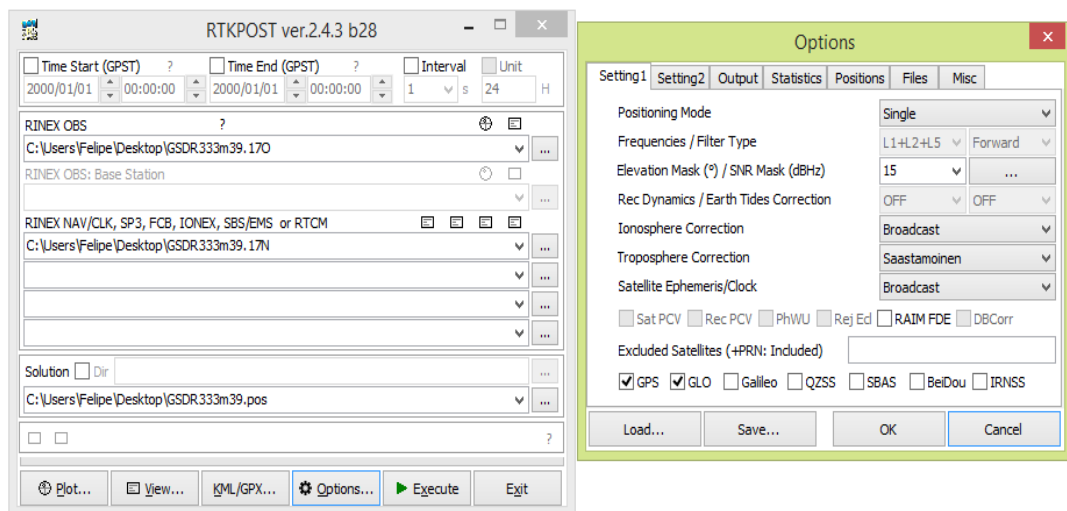
Com esse último comando o software GNSS-SDR é executado e as mensagens dos satélites começam a ser processadas. Para encerrar a execução o comando “q” deve ser dado, e então os arquivos de saída serão disponibilizados no mesmo diretório em que se encontra o arquivo de extensão “.conf” criado.

⁶ Disponível em <http://gnss-sdr.org/conf/>. Acesso em jan. de 2018.

APÊNDICE B: CONFIGURAÇÃO E UTILIZAÇÃO DOS APLICATIVOS RTKPLOT E RTKPOST

Abrindo o aplicativo RTKPost a janela da esquerda da figura abaixo é exibida. Nela, é necessário indicar no campo “RINEX OBS” o arquivo de observação no formato RINEX e o arquivo de navegação no primeiro campo da sessão “RINEX NAV/CLK, SP3, FCB, IONEX, SBS/EMS or RTCM”. Então, no campo “Solution” deve-se informar o caminho onde será disposto o arquivo de saída com os resultados de posicionamento e, por fim, selecionar dentre as opções quais os tipos de correção ionosférica e troposférica, modo de posicionamento, máscara de elevação, como serão obtidos o clock e a efemérides e as constelações que devem ser utilizadas, conforme a figura 8.

Figura 14 - Configurações do RTKPost



Após alguns segundos os resultados de posicionamento são obtidos. Com esses resultados, utilizamos o RTKPlot através do próprio RTKPost, pelo botão “Plot...”, para obtermos a primeira imagem do resultado de posicionamento. Apesar de serem exibidas as coordenadas e a escala na imagem, a fim de obter uma noção mais realista do posicionamento previsto pelo sistema, pode-se utilizar dentro da própria ferramenta um conversor KML/GPX, selecionando o botão “KML/GPX...”, para

que os resultados sejam exibidos com uma imagem do Google Earth, desde que o software esteja instalado no computador.