



Universidade Federal do Rio Grande do Norte (UFRN)

Instituto Metr pole Digital

Natal / RN – Brasil

Tutorial do NT-Pi: implementa  o de um servidor NTP usando RPi e fonte de rel gio GPS

**Diego O. Lemos, Gildo A. S. Freitas, Jeferson F. Ribeiro,
Joaliton L. P. Ferreira, Pedro H. R. Emerick, Valmir C. S. Junior**

**{diegoifr n,gildo.augu,jfrsnr}@gmail.com,
p.emerick@live.com, {luanpereira00,valmircorrea96}@outlook.com**

Este tutorial tem o intuito de auxiliar no processo de configura  o de um servidor NTP Stratum 1 hospedado em um Raspberry Pi, utilizando o m dulo GPS. Ser  um passo-a-passo detalhado desde a conex o do m dulo GPS com os componentes do Raspberry (hardware) at  a configura  o do servidor (software) e com testes de funcionalidade.

Para melhor entendimento dos processos, iremos definir um padr o para esse tutorial. Tamb m dividiremos em subtemas, para que fique mais claro o que ser  feito em cada passo.

Logo abaixo est  o padr o definido para o tutorial:

Explica  o do comando:

```
comando
```

1. CONEXÃO DO GPS NO RASPBERRY PI

Para conectar o Módulo GPS no Raspberry Pi, é necessário fazer as seguintes ligações dos pinos do GPS com os do Raspberry:

GPS	Raspberry Pi
PPS	GPIO24
RXD	TXD
TXD	RXD
GND	GND
VCC	5V

Observação: Tome bastante cuidado ao fazer as ligações para que não danifique o seu microcomputador ou o módulo GPS. Portanto, após as conexões, verifique se está tudo correto.

2. ATUALIZAÇÃO DO RASPBERRY

É importante que o Raspberry Pi esteja atualizado para melhor o uso do seu sistema.

Antes de tudo, verifique se a data e fuso horário do seu Raspberry está correto, senão, defina antes de realizar estes passos.

Apenas por curiosidade, você pode ver a versão do sistema instalado em seu Raspberry:

```
uname -a
```

Realizar o download de pacotes de informações de todas as fontes de configurações:

```
sudo apt-get update
```

Instalar as atualizações disponíveis de todos os pacotes atualmente instalados no sistema, porém remove e instala pacotes também se for necessário:

```
sudo apt-get dist-upgrade
```

Atualizar a firmware do Raspberry Pi:

```
sudo rpi-update
```

3. CONFIGURAÇÕES NO RASPBERRY

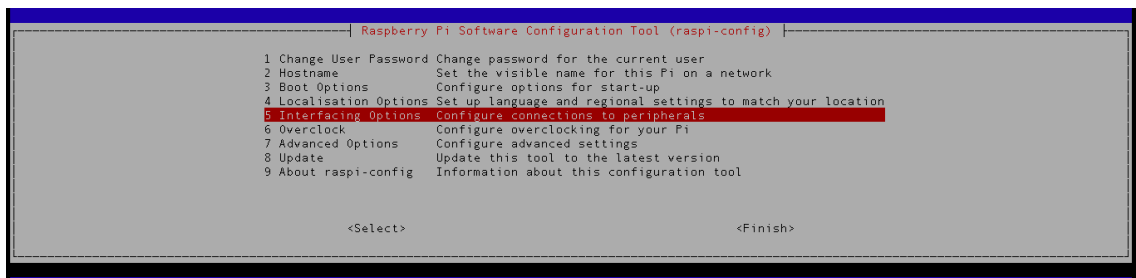
Logo após todas as atualizações realizadas com sucesso, iremos começar a fazer a configuração do Raspberry Pi para que o GPS funcione.

É necessário alterar algumas configurações para que a UART (componente que trata das comunicações seriais) esteja disponível para o GPS, uma vez que é usada para permitir ligar-se a um terminal e autenticar-se. Para isso, iremos seguir os seguintes passos:

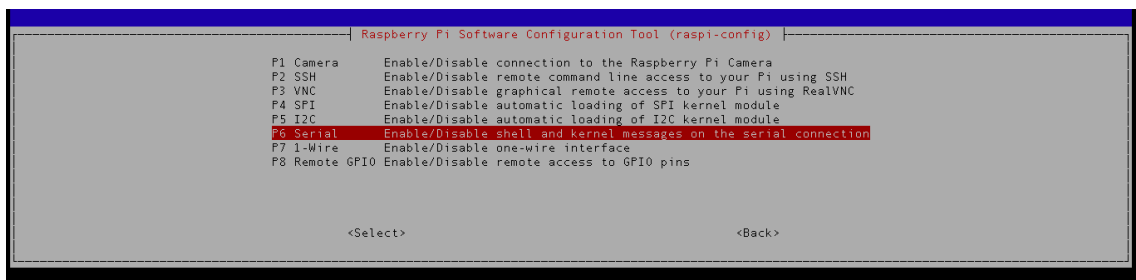
Abra o terminal e digite o comando abaixo para acessar as configurações do Raspberry:

```
sudo raspi-config
```

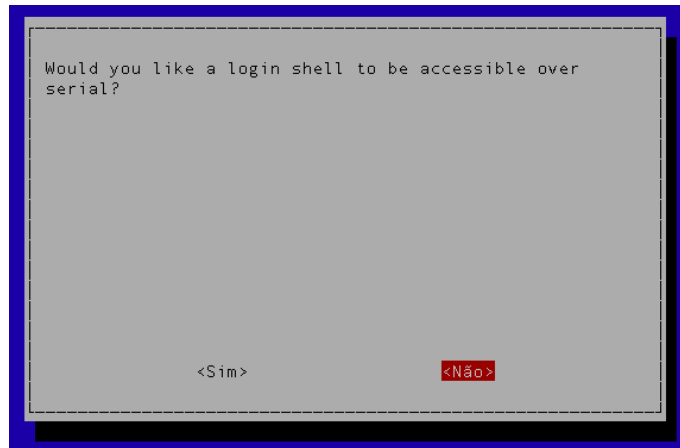
Entre em 'Opções de interface - Configurar conexões para periféricos':



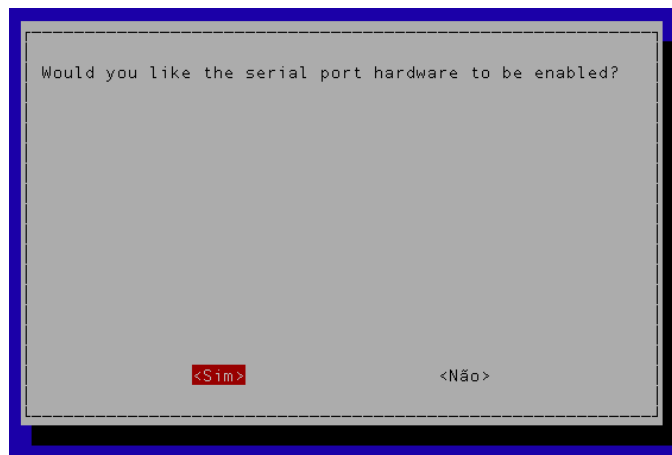
Selecione 'P6 Serial - Ativar / Desativar as mensagens do shell e do kernel na conexão serial':



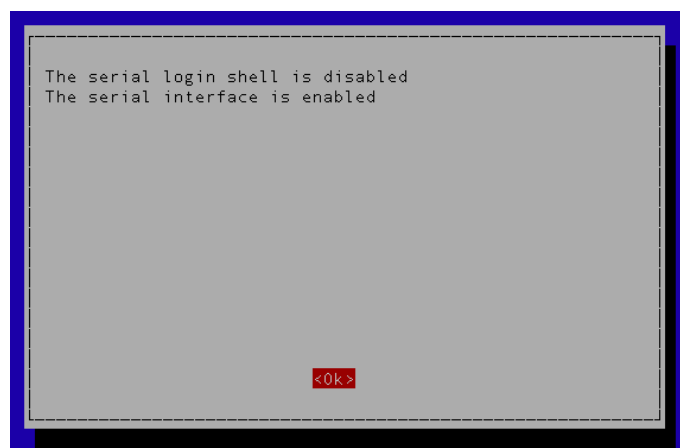
Ao perguntar ‘Gostaria que o shell de login fosse acessível em serial?’ , selecione ‘Não’ e pressione ‘Enter’:



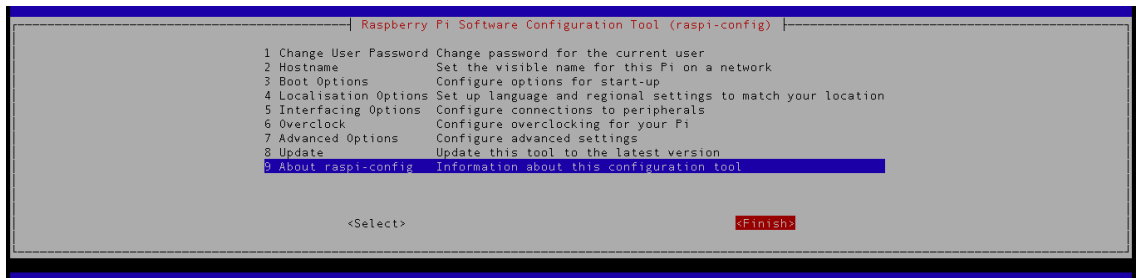
Ao perguntar ‘Gostaria que o hardware da porta serial fosse ativado?’ , selecione ‘Sim’ e pressione ‘Enter’:



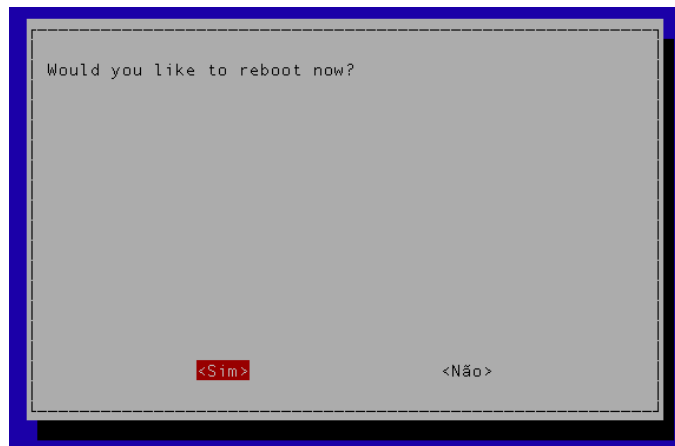
Selecione ‘Ok’ e pressione ‘Enter’:



Selecione 'Finish' e pressione 'Enter', para finalizar as configurações:



Ao perguntar 'Você gostaria de reiniciar agora?', selecione 'Sim' e pressione 'Enter':



4. INSTALAÇÃO E CONFIGURAÇÃO DO GPS E PPS

Agora vamos instalar alguns programas/recursos para melhor utilização do módulo GPS e fazer as configurações necessárias.

Instala o pacote que inclui os cabeçalhos necessários para usar o LinuxPPS (projeto de implementação do PPS) Interface do kernel PPSAPI:

```
sudo apt-get install pps-tools
```

Instala um pacote que implementa recursos que dão um conjunto de privilégios de raiz:

```
sudo apt-get install libcap-dev
```

Instala o GPSD, que é um monitor necessário para coletar as informações do GPS (durante a instalação irá perguntar se deseja continuar com a instalação, basta pressionar 'S' (sim):

```
sudo apt-get install gpsd gpsd-clients python-gps
```

Precisamos colocar o Pi para carregar o módulo do kernel pps-gpio no boot, para isso iremos adicionar no fim do arquivo /etc/modules essa linha 'pps-gpio':

```
sudo sh -c "echo pps-gpio >> /etc/modules"
```

E agora diga para usar GPIO24 como uma entrada PPS, para isso vamos inserir 'dtoverlay=pps-gpio,gpiopin=24' no fim do arquivo /boot/config.txt:

```
sudo sh -c "echo dtoverlay=pps-gpio,gpiopin=24 >> /boot/config.txt"
```

Agora reinicie o Raspberry para validar as configurações:

```
sudo reboot
```

Vamos verificar se o PPS foi configurado corretamente:

```
sudo ppstest /dev/pps0
```

Deverá aparecer mensagens similares a estas:

```
pi@raspberrypi:~$ sudo pps test /dev/pps0
trying PPS source "/dev/pps0"
found PPS source "/dev/pps0"
ok, found 1 source(s), now start fetching data...
source 0 - assert 1511642288.864743547, sequence: 1197 - clear 0.000000000, sequence: 0
source 0 - assert 1511642289.864728737, sequence: 1198 - clear 0.000000000, sequence: 0
source 0 - assert 1511642290.864715750, sequence: 1199 - clear 0.000000000, sequence: 0
source 0 - assert 1511642291.864702034, sequence: 1200 - clear 0.000000000, sequence: 0
source 0 - assert 1511642292.864690923, sequence: 1201 - clear 0.000000000, sequence: 0
source 0 - assert 1511642293.864678248, sequence: 1202 - clear 0.000000000, sequence: 0
source 0 - assert 1511642294.864666355, sequence: 1203 - clear 0.000000000, sequence: 0
source 0 - assert 1511642295.864653837, sequence: 1204 - clear 0.000000000, sequence: 0
^C
```

Para parar as mensagens basta interromper a execução, pressionando ‘Ctrl’ + ‘C’.

Agora iremos configurar o GPSD para isso acesse o arquivo /etc/default/gpsd:

```
sudo nano /etc/default/gpsd
```

Seu arquivo deve estar similar a este abaixo, desconsiderando os comentários marcados pelo ‘#’ no início da linha. Os parâmetros do arquivo deverão ter as seguintes configurações (caso não tenha o parâmetro, adicioná-lo no arquivo):

```
# Inicie o gpsd daemon automaticamente no momento da inicialização
START_DAEMON="true"

# Use o hot-plug USB para adicionar novos dispositivos USB
# automaticamente ao daemon
USBAUTO="true"

# Os dispositivos gpsd devem ser coletados no momento da
# inicialização.
# Eles precisam ser lidos/graváveis, seja pelo usuário gpsd ou pelo
# grupo de discagem.
# No nosso caso o dispositivo GPS era o ttyAMA0
DEVICES="/dev/ttyAMA0"

# Outras opções que você deseja passar para gpsd:
# Observe que você precisará fazer o gpsd começar automaticamente no
# momento da inicialização e para indicar à ferramenta de configuração
# o dispositivo a ser usado e adicionar a opção "-n" para trabalhar
# com o NTP
GPSD_OPTIONS="-n"
```



```
# Define um soquete de controle
GPSD_SOCKET="/var/run/gpsd.sock"
```

Após fazer as devidas configurações, pressione ‘Ctrl’ + ‘X’ para sair, ‘S’ e depois ‘Enter’ para salvar.

Agora reinicie o Raspberry para validar as configurações:

```
sudo reboot
```

Crie um soquete de controle para adição de dispositivo e comandos de remoção. Você deve especificar um nome de caminho válido em seu sistema de arquivos local. Isso será criado como um soquete de domínio Unix ao qual você pode escrever comandos que editam a lista de dispositivos internos do daemon:

```
sudo gpsd /dev/ttyAMA0 -F /var/run/gpsd-sock
```

Para ver as informações que o GPS manda para o Raspberry:

```
sudo cat /dev/ttyAMA0
```

Deverá aparecer mensagens similares a estas:

```
[pi@raspberrypi:~ $ sudo cat /dev/ttyAMA0
$GPRMC,210430.00,A,0550.65031,S,03512.79380,W,0.658,,251117,,,A*70
$GPVTG,,T,,M,0.658,N,1.219,K,A*23
$GPGGA,210430.00,0550.65031,S,03512.79380,W,1,06,1.39,58.0,M,-10.2,M,,*48
$GPGSA,A,3,25,18,12,29,15,10,,,,,,,,,3.02,1.39,2.68*07
$GPGSV,4,1,13,05,04,145,20,10,26,318,22,12,37,021,16,13,13,101,*71
$GPGSV,4,2,13,15,37,076,21,18,56,305,18,20,51,158,07,21,30,222,*75
$GPGSV,4,3,13,24,09,026,,25,70,335,31,26,04,219,18,29,43,153,22*78
$GPGSV,4,4,13,31,05,270,*49
$GPGLL,0550.65031,S,03512.79380,W,210430.00,A,A*63
$GPGST,210430.00,50,,,,,14,14,20*7A
$GPZDA,210430.00,25,11,2017,00,00*61
$GPGBS,210430.00,13.7,13.6,20.2,,,,,*74
$GPTXT,01,01,01,NMEA unknown msg*58
```

Para parar as mensagens basta interromper a execução, pressionando ‘Ctrl’ + ‘C’.

Agora podemos ver os dados que chegam ao gpssd (as vezes o gps não consegue se comunicar com algum satélite, pode ser pelo tempo nublado, ou falta de sinal por estar em um local fechado, para melhor funcionamento coloque o seu Raspberry com o gps próximo de um local aberto, por exemplo janelas):

```
cgps -s
```

Deverá aparecer uma mensagem similar a esta:

Time: 2017-11-25T20:26:27.000Z	PRN: 5	Elev: 18	Azim: 143	SNR: 19	Used: N
Latitude: 5.844156 S	10	12	325	19	N
Longitude: 35.213054 W	12	55	029	24	Y
Altitude: 88.7 m	13	13	083	17	Y
Speed: 0.2 kph	15	33	055	27	Y
Heading: 137.5 deg (true)	18	41	322	30	Y
Climb: 9.2 m/min	20	69	152	16	Y
Status: 3D FIX (17 secs)	21	30	242	24	Y
Longitude Err: +/- 16 m	24	15	012	16	N
Latitude Err: +/- 11 m	29	34	172	26	Y
Altitude Err: +/- 43 m	31	04	253	20	N
Course Err: n/a					
Speed Err: +/- 2 kph					
Time offset: 0.007					
Grid Square: HI24jd					

Para sair desta tela basta interromper a execução, pressionando 'Ctrl' + 'C'.

5. INSTALAÇÃO DO NTP

Instale o NTP fornecida pelo Pi:

```
sudo apt-get install ntp
```

A versão do ntpd fornecida com o Pi não inclui suporte PPS. Então, devemos compilar o nosso através das fontes.

É necessário habilitar os repositórios de fontes, para isso entre no arquivo `/etc/apt/sources.list` e descomente a linha que contém `'deb-src'`, basta apagar o `'#'` do início da linha:

```
sudo nano /etc/apt/sources.list
```

Agora para obter as listas de pacotes:

```
sudo apt-get update
```

Nós não vamos obter a fonte para o pacote ntp desta forma, mas precisamos das ferramentas de compilação necessárias, então instale-os:

```
sudo apt-get build-dep ntp
```

Crie uma pasta para o download do ntp e entre dentro da pasta criada:

```
mkdir ntp  
cd ntp
```

Agora iremos fazer o download do ntp. Verifique no site do NTP <http://www.ntp.org/downloads.html> qual o link para o download da versão mais recente. No desenvolvimento deste tutorial, a versão mais recente era a 4.2.8p.10:

```
wget https://www.eecis.udel.edu/~ntp/ntp_spool/ntp4/ntp-4.2/ntp-4.2.8p10.tar.gz
```

Descompacte o arquivo baixado e entre na pasta criada:

```
tar xvfz ntp-4.2.8p10.tar.gz  
cd ntp-4.2.8p10/
```

Então rode o script de configuração. Se estiver usando o Raspberry 2/3 use o comando de configuração abaixo, senão, retire a flag ‘--enable-linuxcaps’:

```
./configure --enable-linuxcaps --prefix=/usr
```

Compile e instale. Se estiver usando o Raspberry 2/3 e ele esteja conectado a uma fonte de alimentação 2A, use o comando de compilação abaixo, senão, retire a flag ‘-j5’:

```
make -j5  
sudo make install
```

6. CONFIGURAÇÃO DO NTP

Crie o arquivo ntp.drift:

```
sudo touch /etc/ntp.drift
```

Agora vamos configurar o ntpd, para isso acesse seu arquivo de configuração:

```
sudo nano /etc/ntp.conf
```

O seu arquivo deve estar com o seguinte conteúdo:

```
# "Memoria" para o escorregamento de frequencia do micro
# pode ser necessario criar esse arquivo manualmente com
# o comando touch ntp.drift
driftfile /var/lib/ntp/ntp.drift

# Estatisticas do ntp que permitem verificar o historico de
# funcionamento e gerar graficos
statsdir /var/log/ntpstats/
statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable

# Configuracao do servidor GPS e PPS
server 127.127.28.0 minpoll 4 maxpoll 4
fudge 127.127.28.0 time1 0.000 refid GPS
server 127.127.22.0 minpoll 4 maxpoll 4
fudge 127.127.22.0 flag3 1 refid PPS

# GPS with PPS enabled
server 127.127.20.0 mode 17 minpoll 4 maxpoll 4 iburst true
fudge 127.127.20.0 flag1 1 refid GPS

# Configuracoes de restricao de acesso
restrict default kod nomodify notrap nopeer noquery
restrict -6 default kod nomodify notrap nopeer noquery
```

```
# Os usuários locais podem interrogar mais o servidor ntp.
restrict 127.0.0.1
restrict ::1

# Desabilitar comando monlist
disable monitor
```

Após fazer as devidas modificações, pressione ‘Ctrl’ + ‘X’ para sair, ‘S’ e depois ‘Enter’ para salvar.

Para melhor entendimento das configurações feitas para o servidor com o GPS, é recomendado acessar os links abaixo, onde todos os servidores usados e flags são explicadas detalhadamente:

- <https://www.eecis.udel.edu/~mills/ntp/html/drivers/driver28.html>
- <https://www.eecis.udel.edu/~mills/ntp/html/drivers/driver22.html>
- <https://www.eecis.udel.edu/~mills/ntp/html/drivers/driver20.html>

Reinicie o serviço ntp:

```
sudo service ntp restart
```

Antes de executar este comando, aguarde alguns minutos para o ntp começar a funcionar verdadeiramente, e execute o comando abaixo para ver o ntp funcionando:

```
ntpq -pn
```

Deverá aparecer uma mensagem similar a esta, caso contrário, significa que o serviço ntp não está funcionando:

```
pi@raspberrypi:~ $ ntpq -pn
      remote           refid      st t when poll reach   delay   offset  jitter
=====
x127.127.28.0      .GPS.           0 1  14  16  377    0.000    0.134   1.172
x127.127.22.0      .PPS.           0 1  13  16  377    0.000   134.403   0.909
```

Explicação sobre cada parâmetro:

- Remote: apresenta o nome ou IP do servidor de horas;
- Refid: a referência que o servidor está utilizando para sincronizar.
- St: estrato da fonte de tempo;

- When: apresenta o tempo em segundos, que se passaram desde a última consulta a fonte de horas;
- Poll: frequência de segundos que a fonte é consultada;
- Reach: é um número octal que indica o resultado das últimas 8 tentativas de acesso ao servidor. O valor 377 indica que as últimas 8 consultas ao servidor obtiveram sucesso. Outros valores indicam falhas;
- Delay: significa o tempo de ida e volta dos pacotes. O valor do atraso não é tão importante, mas o ntp é sensível à simetria da rede. Ou seja, o ntp espera que o tempo de ida dos pacotes seja igual ao da volta. Isso geralmente está relacionado a atrasos pequenos.
- Offset: significa o quanto o relógio local tem de ser adiantado ou atrasado para se igualar ao relógio do servidor. Quanto menor o valor, melhor. Valores de alguns milissegundos podem ser considerados normais.
- Jitter: é calculada à partir de diversas medidas de deslocamento. Representa o quanto essas medidas variam entre si. Valores pequenos são melhores. Valores grandes podem indicar problemas de conectividade.

Caso queira ver mais informações sobre o seu ntp:

```
ntpq -c rl
```

Deverá aparecer uma mensagem similar a esta:

```
[pi@raspberrypi:~ $ ntpq -c rl
associd=0 status=c418 leap_alarm, sync_uhf_radio, 1 event, no_sys_peer,
version="ntpd 4.2.8p10@1.3728 Fri Nov 17 18:39:58 UTC 2017 (1)",
processor="armv7l", system="Linux/4.9.41-v7+", leap=11, stratum=1,
precision=-19, rootdelay=0.000, rootdisp=74.686, refid=GPS,
reftime=ddc45adb.dc826379 Sat, Nov 25 2017 18:10:51.861,
clock=ddc45add.60c6b37b Sat, Nov 25 2017 18:10:53.378, peer=15053, tc=4,
mintc=3, offset=0.000000, frequency=0.000, sys_jitter=0.000000,
clk_jitter=65.855, clk_wander=0.000
```

Explicação sobre alguns parâmetros:

- Version: versão do ntp;
- Stratum: estrato do servidor local.
- Precision: indica a precisão do servidor com o expoente de um número base 2;
- Rootdelay: representa o atraso dos pacotes em milissegundos;
- Rootdisp: representa o erro máximo da medida de offset em relação ao estrato 0;

- Refid: principal referência do sistema;
- Offset: representa quanto o relógio local precisa ser atrasado ou adiantado até atingir o horário correto;
- Frequency: representa o erro na frequência do relógio local.

7. CONCLUSÃO

O seu servidor NTP já está funcionando e pode ser usado em sua rede local através do IP do Raspberry. O site NTP.br disponibiliza tutoriais de como fazer a configuração do NTP em sua máquina, basta que, ao pedir um servidor para sincronização, você coloque o IP do seu Raspberry.

Links:

- Configuração Windows - <https://ntp.br/guia-win-comum.php>
- Configuração Linux - <https://ntp.br/guia-linux-comum.php>
- Configuração Mac - <https://ntp.br/guia-mac.php>

8. REFERÊNCIAS

Alteageek, tutoriales, raspberry pi y cisco, en español. Sem data. Disponível em: <<https://alteageek.com/instalar-un-gps-en-una-raspberry-pi/>>. Acesso em: 11 nov. 2017.

Alteageek, tutoriales, raspberry pi y cisco, en español. Sem data. Disponível em: <<https://alteageek.com/2016/01/31/convertir-una-raspberry-pi-1-en-servidor-ntp-stratum-1/>>. Acesso em: 11 nov. 2017.

CATB. Sem data. Disponível em: <<http://www.catb.org/gpsd/gpsd.html>>. Acesso em: 11 nov. 2017.

Mundo FreeBSD. Bem Vindo ao nosso mundo!. Servidor NTP no FreeBSD. 2016. Disponível em: <<http://mundofreebsd.com.br/servidor-ntp-no-freebsd/>>. Acesso em: 11 nov. 2017.

Network Time Foundation. NTP: The Network Time Protocol. 2017. Disponível em: <<http://www.ntp.org/>>. Acesso em: 11 nov. 2017.

NTP.BR. Guia Linux/BSD. Sem data. Disponível em: <<https://ntp.br/guia-linux-avancado.php>>. Acesso em: 11 nov. 2017.

SETTING UP A STRATUM 1 NTP SERVER ON A RASPBERRY PI. 2016. Disponível em: <<https://frillip.com/raspberry-pi-stratum-1-ntp-server/>>. Acesso em: 11 nov. 2017.

The Raspberry Pi as a Stratum-1 NTP Server. 2017. Disponível em: <<http://www.satsignal.eu/ntp/Raspberry-Pi-NTP.html>>. Acesso em: 11 nov. 2017.