Hands-on 6

Parte 02: Transmissor WBFM (GRC)

**Exercício**

**OBJETIVO:** Através do uso de conceitos sobre modulação em frequência (FM) e de alguns blocos básicos mostrados em tutoriais passados iremos, com a ajuda do GNR (*GNU Radio Companion*), criar um transmissor FM capaz de enviar ao espaço livre áudio proveniente de arquivo “.*wav*” ou microfone.

1. Caso ainda não esteja aberto, inicialize o GNU Radio Companion.
   1. Abra um terminal digitando CRTL+ALT+t e digite: gnuradio-companion e pressione ENTER

|  |
| --- |
| fig_01.png |

Alternativa:

1. Clique em Dash Home e digite gnuradio e clique no ícone correspondente ao GRC

|  |
| --- |
|  |

1. Com o GNU RADIO COMPANION aberto, crie um novo projeto.

|  |
| --- |
| fig04.png |

1. Clique duas vezes no Bloco ***Options***. Esse bloco configura alguns parâmetros gerais de flowgraph. Mantenha o *ID* como top\_block. Digite um título para o projeto e um autor. Selecione *Generate Options* com *QT GUI*, *Run* para *Autostart* e *Realtime Scheduling* para *Off*. Então, feche a janela de propriedades.
2. Construa um projeto utilizando os blocos ***WavFileSource****, dois* ***MultiplyConst****, dois* ***QT GUI Range****, um* ***WBFM Transmit*e *UHD: USRP Sink***. Conecte os elementos de forma que sua área de trabalho fique igual a figura a seguir. Altere o campo *Type* para *Float* apenas do bloco ***MultiplyConst*** que estará ligado à saída do bloco ***Wav File Source***.

|  |
| --- |
|  |

1. Configure os dois blocos ***QT GUI Range*** para controlar o valor das constantes dos blocos ***Multiply Const***. Altere o ID dos blocos ***QT GUI Range*** para *slider\_0* e *slider\_1*, respectivamente. Altere o campo *Default Value* para 1, mude o valor dos campos *Start*, *Stop*e *Step* para 0, 10 e 0.1, respectivamente. Além disso, deixe o campo *Widget* na opção *Counter + Slider.* Os blocos devem ficar como mostrado abaixo.

|  |  |
| --- | --- |
|  |  |

1. Agora configure o bloco ***MultiplyConst*** que se encontra ligado ao bloco ***Wav File Source.*** No campo *Constant* altere o valor para *slider\_0*(veja figura a seguir). Faça a mesma configuração no bloco ***MultiplyConst*** que se encontra ligado a saída do bloco ***WBFM Transmit***, mudando o valor do campo *Constant* para *slider\_1*.

|  |  |
| --- | --- |
|  |  |

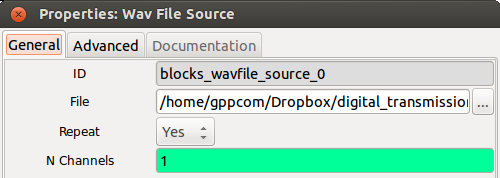
1. Agora vamos configurar o bloco ***WBFM Transmit***. Altere o campo ***Audio Rate*** para taxa de amostragem de sua placa de som, que por padrão deve ser 44100. O campo *Quadrature Rate* está diretamente ligado ao campo *Audio Rate, no qual* só é permitido colocar valores múltiplos do valor que você colocou no campo *Audio Rate*. No nosso caso, colocaremos 882000 que é um valor 20 vezes maior que 44100. O campo *Tau* e *Max Deviation* deixaremos o valor que vem padrão. O *Tau* remete ao valor da constante de tempo de um circuito RC, enquanto o *Max Deviation* é a frequência máxima de desvio determinada pela Agência reguladora (ANATEL), que é de 75KHz. O bloco deve ficar como na figura a seguir.

|  |
| --- |
|  |

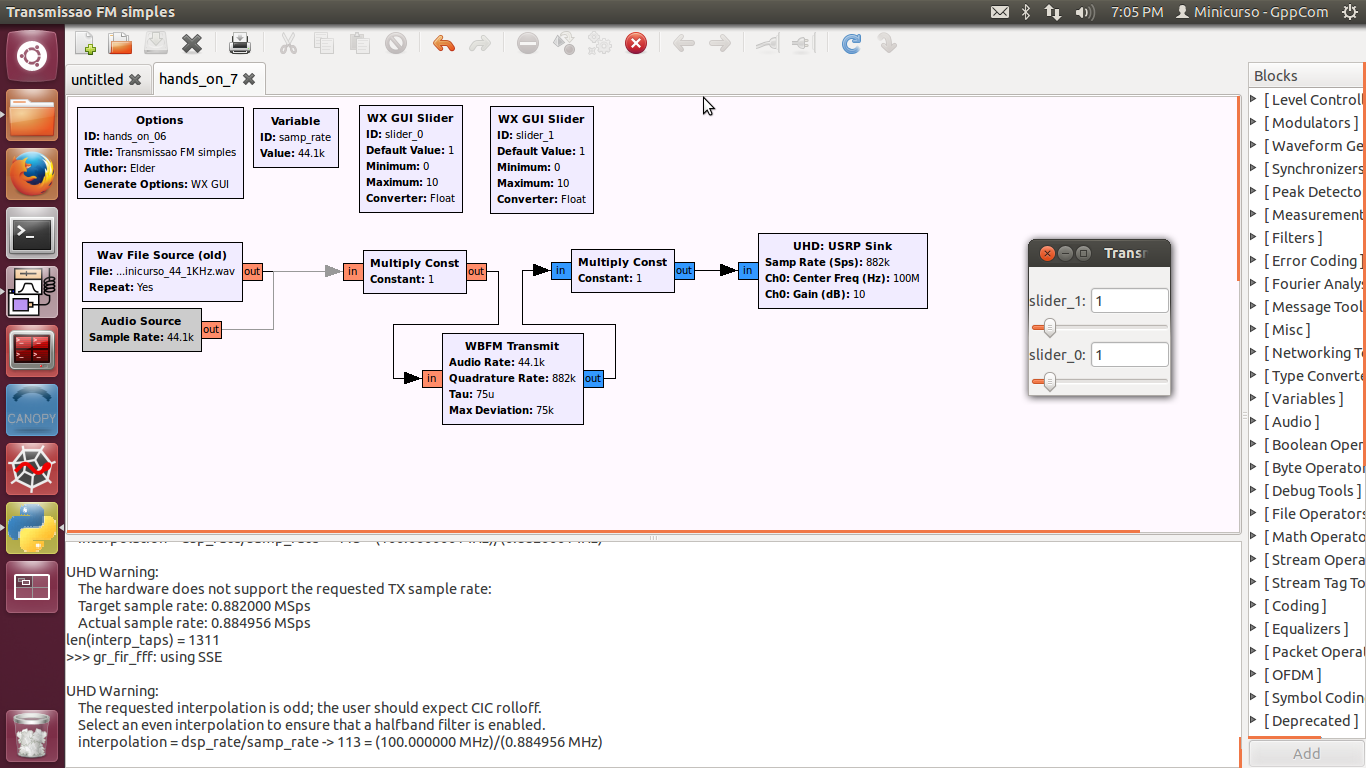
1. Dê dois cliques no bloco ***UHD: USRP Sink*** e altere o campo *Ch0:Center Freq* para a frequência que deseja alocar a transmissão. No nosso caso optaremos pela frequência de 88,9MHz. Agora altere o campo *Ch0:Gain(dB)* que é o valor de ganho da antena para 10. Finalmente altere o campo *Samp Rate (Sps)* para o valor da taxa de quadratura que colocamos no bloco ***WBFM Transmit*** que foi de 882000. O bloco deve ficar como na figura a seguir.

|  |
| --- |
|  |

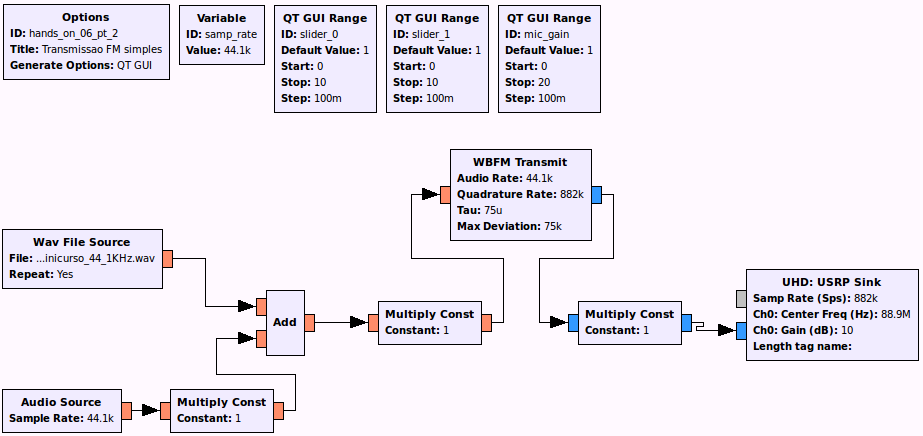
1. Dê dois cliques no bloco ***Wav File Source*** e procure um arquivo wav (no nosso caso, procure na área de trabalho o arquivo chamado “**handson2\_file\_minicurso\_44\_1KHz.wav**”. Selecione o arquivo **wav** com 44.1 kHz de taxa de amostragem. O bloco deve ficar como na figura a seguir.



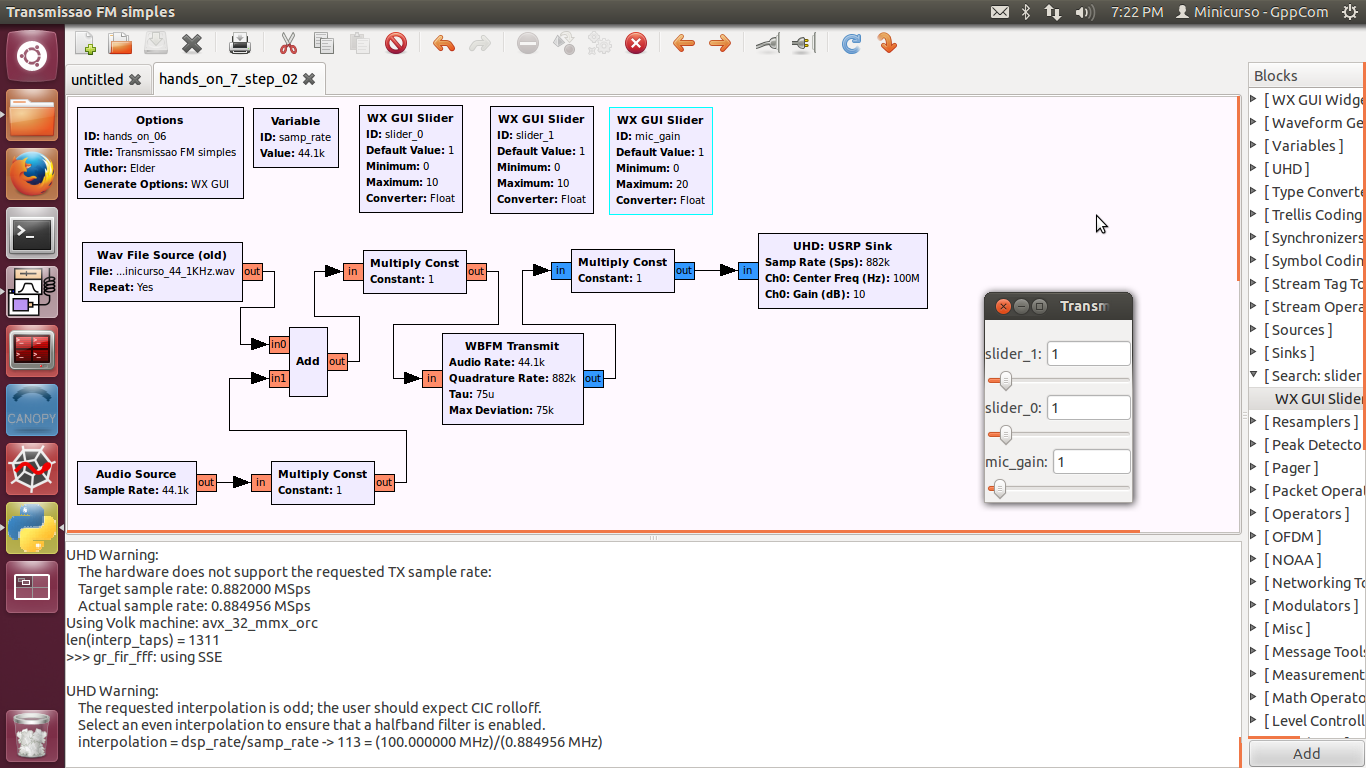
1. Salve o projeto com o nome ***hands\_on\_6.grc*,** gere o projeto em Python e execute-o. Agora tente sintonizar um receptor FM comercial (ou o seu celular) em 88,9 MHz. Você ouvirá a musica do arquivo .wav. Também será mostrada uma tela como na figura a seguir. Os sliders podem ser usados para ajustar o ganho do sinal antes e depois do transmissor.



1. Também podemos usar o áudio do microfone como entrada do transmissor. Pare a execução anterior e salve o projeto como “**hands\_on\_6\_step\_02.grc**”. Arraste os seguintes blocos para sua área de trabalho: um bloco ***AudioSource,*** um bloco **Add,** outro bloco ***Multiply Constant***. Eles devem ser conectados como na figura a seguir.



1. Configure o novo bloco ***QT GUI Range*** para controlar o valor da constante do novo ***Multiply Const***. Altere o ID do novo bloco ***QT GUI Range*** para *mic\_gain*. Altere o campo *Default Value* para 1, mude o valor dos campos *Minimum* e *Maximum* para 0 e 20, respectivamente. Agora configure o bloco ***MultiplyConst*** que se encontra ligado ao bloco ***AudioSource***. No campo *Constant* altere o valor para *mic\_audio*.
2. Gere e execute o projeto. O sinal transmitido será um sinal de áudio recebido pelo driver da placa de som (microfone) misturado com a música do arquivo WAV. Se não tiver ouvindo a sua voz ao falar no microfone, aumento o ganho no slider novo (mic\_gain). Ele deve aparecer como a figura a seguir.



**Referências**

[1] http://www.snowymtn.ca/GNURadio/GNURAdioDoc-7.pdf - acesso em:04/12/2012