

SAE22 - Mesurer et caractériser un signal ou un système

1 - Introduction

Auto-évaluation de groupe

NOM Prénom	Victoria Ruf	Thomas Godet	Bastian Pitois	Ho Koloina Raveloson	Florian Vigneux	Lemée Raphaël
Recherches, théorie	A	B	A	A	B	C
Rédaction	A	B	A	A	A	C
Comportement, sérieux	A	A	A	A	A	C

Dans le cadre de la SAE22, nous avons réalisé un TP visant à mesurer et analyser le spectre radiofréquence, en particulier celui de la téléphonie mobile. À l'aide d'un récepteur RTL-SDR et d'un script MATLAB, nous avons capté les signaux dans la bande FM, puis étudié et corrigé les données spectrales obtenues. Ce travail nous a permis de mettre en pratique les notions de traitement du signal et de mieux comprendre l'organisation du spectre des fréquences en France.

2 - Étude du spectre radiofréquence

2.1 - Allez sur le site de l'ANFR pour récupérer le poster "organisation du spectre des fréquences" avec tous les services télécom en France. prenez le temps de la lire et d'y repérer les services télécom les plus connus

Radiodiffusions commerciales :

- FM : 87.5 - 108 MHz
- Grandes ondes : 148.5 - 255 kHz
- Moyennes ondes : 526.5 - 1606.5 kHz
- Télévision Numérique terrestre : 470 - 694 MHz

Téléphonie mobile :

- 2G : 900 - 1800 MHz
- 3G : 900 - 2100 MHz
- 4G : 700 - 2600 MHz
- 5G : 3 bandes principales :
 - 700 MHz
 - 3,4 - 3,8 GHz → bande cœur
 - 26 GHz → bande millimétrique

Le "G" signifie génération.

2.2 - Lancer l'exécution du fichier `rtlsdr_rx_specsweep.m` dans Matlab, ne plus toucher à l'antenne Observer l'espace Command Window de Matlab pour comprendre ce que réalise ce fichier et attendre la fin de son exécution

```

function rtl_sdr_rx_specsweep

% PARAMETERS (can change)
location      = 'RT-FM-DateDuJour'; % location used for figure name
start_freq    = 86e6; % sweep start frequency
stop_freq     = 109e6; % sweep stop frequency
rtl_sdr_id    = '0'; % RTL-SDR stick ID
rtl_sdr_fs    = 2.8e6; % RTL-SDR sampling rate in Hz
rtl_sdr_gain   = 40; % RTL-SDR tuner gain in dB
rtl_sdr_fmrlen = 4096; % RTL-SDR output data frame size
rtl_sdr_datatype = 'single'; % RTL-SDR output data type
rtl_sdr_ppm    = 0; % RTL-SDR tuner parts per million correction
% PARAMETERS (can change, but may break code)
nfft           = 20; % number of frames to receive
fft_hold       = 'avg'; % hold function "max" or "avg"
nfft           = 4096; % number of points in FFTs (2^something)
dec_factor     = 16; % output plot downsample
overlap        = 0.5; % FFT overlap to counter rolloff
nfft           = 100; % number of frames to dump after retuning (to clear buffer)

% CALCULATIONS
rtl_sdr_tunerfreq = start_freq:rtl_sdr_fs*overlap:stop_freq; % range of tuner frequency in Hz
if (max(rtl_sdr_tunerfreq) < stop_freq) % check the whole range is covered, if not, add an extra tuner freq
    rtl_sdr_tunerfreq(length(rtl_sdr_tunerfreq)+1) = max(rtl_sdr_tunerfreq)+rtl_sdr_fs*overlap;
end
nretunes = length(rtl_sdr_tunerfreq); % calculate number of retunes required
freq_bin_width = (rtl_sdr_fs/nfft); % create xaxis
freq_axis = (rtl_sdr_tunerfreq(1)-rtl_sdr_fs/2*overlap : freq_bin_width*dec_factor : (rtl_sdr_tunerfreq(end)+rtl_sdr_fs/2*overlap)-freq_bin_width)/1e6;

% create spectrum figure
h_spectrum = create_spectrum;

% run capture and plot
capture_and_plot;

% make spectrum visible
h_spectrum.fig.Visible = 'on';

% save data
filename = ['rtl_sdr_rx_specsweep_', num2str(start_freq/1e6), 'MHz_', num2str(stop_freq/1e6), 'MHz_', location, '.fig'];
savefig(filename);

```

- Le script fait un balayage du spectre radio en changeant progressivement la fréquence centrale du récepteur RTL-SDR. Il commence à 86 MHz et augmente d'environ 1,4 MHz : $2,8 \text{ MHz} \times 0,5 = 1,4 \text{ MHz}$ où 2,8 MHz est la fréquence d'échantillonnage (rtl_sdr_fs) et 0,5 l'overlap.
- La dernière fréquence centrale utilisée est 109,8 MHz (fc = 109,8MHz), ce qui dépasse légèrement la plage 86-109 MHz (stop_freq et start_freq) pour être sûr d'avoir la plage complète.

```

>> rtl_sdr_rx_specsweep

fc = 86MHz
fc = 87.4MHz
progress = 10%
fc = 88.8MHz
fc = 90.2MHz
progress = 20%
fc = 91.6MHz
fc = 93MHz
progress = 30%
fc = 94.4MHz
fc = 95.8MHz
progress = 40%
fc = 97.2MHz
progress = 50%
fc = 98.6MHz
fc = 100MHz
progress = 60%
fc = 101.4MHz
fc = 102.8MHz
progress = 70%
fc = 104.2MHz
fc = 105.6MHz
progress = 80%
fc = 107MHz
fc = 108.4MHz
progress = 90%
fc = 109.8MHz
progress = 100%

run time = 9.551s

```

2.3 - Traduisez en Français et analysez la légende de la figure obtenue : vous commenterez séparément chaque élément de cette légende

- **RTL-SDR Spectrum Sweep** : Balayage du spectre avec le récepteur RTL-SDR (=antenne)
- **Range = 86MHz to 109MHz** : Plage de fréquence analysée, de 86 MHz à 109 MHz (=la FM)
- **Bin Width = 10.9375kHz** : Largeur de chaque division fréquentielle, soit 10,9375 kHz
- **Number of Bins = 2304** : Nombre total de divisions fréquentielles dans l'analyse
- **Number of Retunes = 18** : Nombre de réajustements de fréquence du récepteur RTL-SDR pendant le balayage

Le graphique supérieur présente le "Power Ratio (dBm) [relative to 50 Ω load]" (Ratio de puissance en dBm relatif à une charge de 50 Ω), et le graphique du dessous affiche la "Relative Power (Watts)" (Puissance relative en Watts) avec en abscisse "x" la Fréquence (MHz).

2.4 - Analysez le code Matlab du fichier pour retrouver les variables liées à cette légende

XXIII - Présentez explicitement toutes les variables associées aux éléments de la légende

Les variables clés liées à la légende sont :

- `start_freq = 86e6` : Fréquence de début en Hz (86 MHz)
- `stop_freq = 109e6` : Fréquence de fin en Hz (109 MHz)
- `freq_bin_width = (rtlsdr_fs/nfft)*dec_factor` : Largeur de chaque bin (calculée à partir de la fréquence d'échantillonnage, du nombre de points FFT et du facteur de décimation)
- `nretunes = length(rtlsdr_tunerfreq)` : Nombre de réajustements de fréquence centrale
- `freq_axis` : Vecteur de fréquences converti en MHz
- `nfft` : Nombre de points FFT
- `ntune` : Est une variable qui sert de compteur pour parcourir les différentes fréquences centrales du RTL-SDR pendant le balayage du spectre.

XXIV - Donner l'unité et la nature de `fft_masterreshape` d'après la formule utilisée pour afficher les spectres en fréquence et/ou son calcul initial

La variable `fft_masterreshape` est obtenue à partir de : `fft_masterreshape = reshape(fft_dec',1,ntune*nfft*overlap/dec_factor);`

Représente l'amplitude spectrale en volts (V) car elle provient de la transformée de Fourier des signaux temporels.

XXV - Quelle est la valeur de l'impédance utilisée dans ce calcul ?

L'impédance utilisée est de 50 Ω , comme on peut le voir dans la formule de conversion en dBm et dans la légende du graphique : `y_data_dbm = 10*log10((fft_masterreshape.^2)/50);`

XXVI - Retrouvez la ligne de code permettant le calcul des amplitudes de la courbe en dBm : De quelle grandeur s'agit-il ? Donnez la formule théorique correspondante et expliquez l'erreur faite par l'auteur de ce fichier à partir de la formule théorique

La ligne de code permettant le calcul des amplitudes de la courbe en dBm :

`y_data_dbm = 10*log10((fft_masterreshape.^2)/50);`

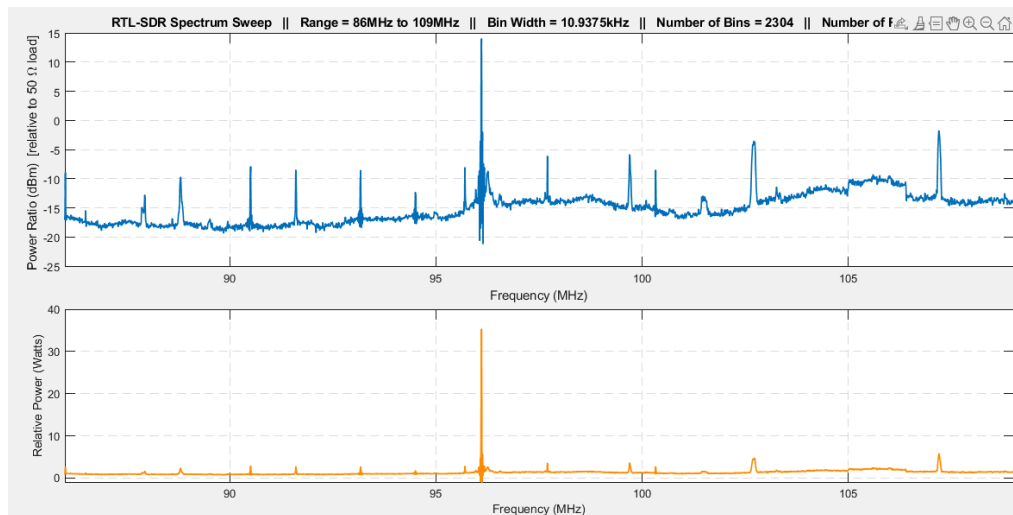
La formule théorique pour convertir une amplitude V en dBm est : $P(\text{dBm}) = 10 \cdot \log_{10}(V^2/R)$

Où R est l'impédance (50 Ω). Or ici l'erreur faite par l'auteur c'est que il y a pas de conversion de watts en milliwatts, d'où il faut diviser la puissance obtenue dans la parenthèse (Pw) par 10e-3.

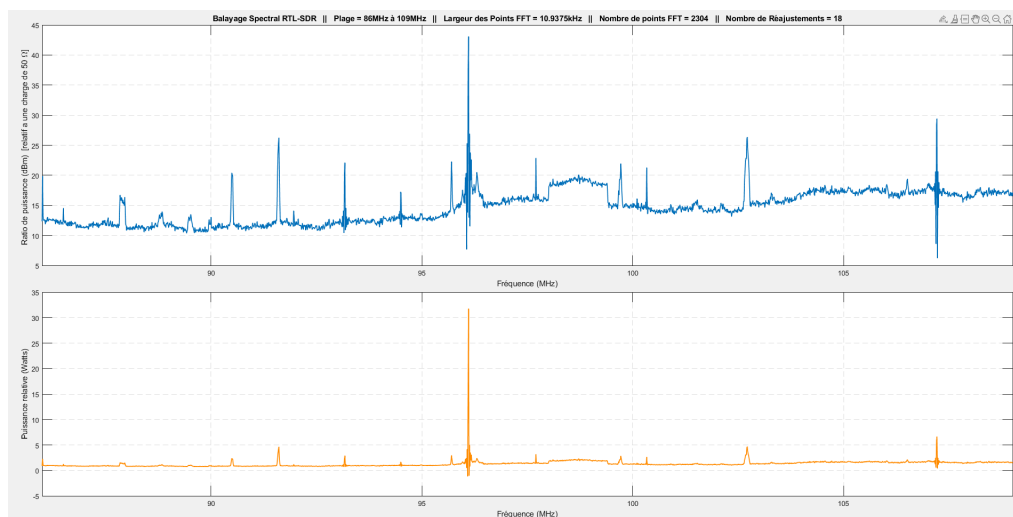
Ainsi on obtient : `y_data_dbm = 10*log10((fft_masterreshape.^2)/50/0.001);`

XXVII - Implantez-la dans le script Matlab et testez votre solution en relevant un spectre final ainsi corrigé

Avant modification :



Après modification :



XXVIII - Retrouvez les lignes de code réalisant les tracés des 2 courbes et les modifier pour écrire une légende entièrement en français avec les bonnes unités !

Les lignes de code réalisant les tracés sont :

```

90 % subplot 1
91 h_spectrum.axes1 = axes(...
92     'Parent',h_spectrum.fig,...
93     'YGrid','on','YColor',h_spectrum.axes_grey,...
94     'XGrid','on','XColor',h_spectrum.axes_grey,...
95     'GridLineStyle','--',...
96     'Color',h_spectrum.plot_white);
97 box(h_spectrum.axes1,'on');
98 hold(h_spectrum.axes1,'on');
99 xlabel(h_spectrum.axes1,'Frequency (MHz)');
100 ylabel(h_spectrum.axes1,'Power Ratio (dBm) [relative to 50 \Omega load] ');
101 xlim(h_spectrum.axes1,[start_freq/1e6,stop_freq/1e6]);
102
103 % subplot 2
104 h_spectrum.axes2 = axes(...
105     'Parent',h_spectrum.fig,...
106     'YGrid','on','YColor',h_spectrum.axes_grey,...
107     'XGrid','on','XColor',h_spectrum.axes_grey,...
108     'GridLineStyle','--',...
109     'Color',h_spectrum.plot_white);
110 box(h_spectrum.axes2,'on');
111 hold(h_spectrum.axes2,'on');
112 xlabel(h_spectrum.axes2,'Frequency (MHz)');
113 ylabel(h_spectrum.axes2,'Relative Power (Watts)');
114 xlim(h_spectrum.axes2,[start_freq/1e6,stop_freq/1e6]);
115
116 % figure title
117 title(h_spectrum.axes1,['RTL-SDR Spectrum Sweep || Range = ',num2str(start_freq/1e6),'MHz to ',...
118     num2str(stop_freq/1e6),'MHz || Bin Width = ',num2str(freq_bin_width*dec_factor/1e3),...
119     'kHz || Number of Bins = ',num2str(length(freq_axis)), ' || Number of Retunes = ',...
120     num2str(nretunes)]);

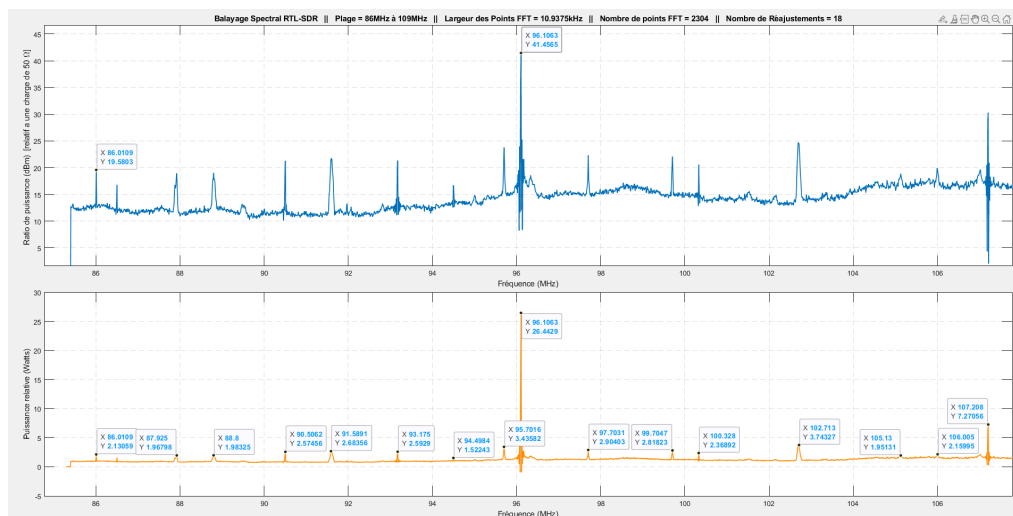
```

XXIX - Quelle est la fréquence d'échantillonnage de cette grandeur ?

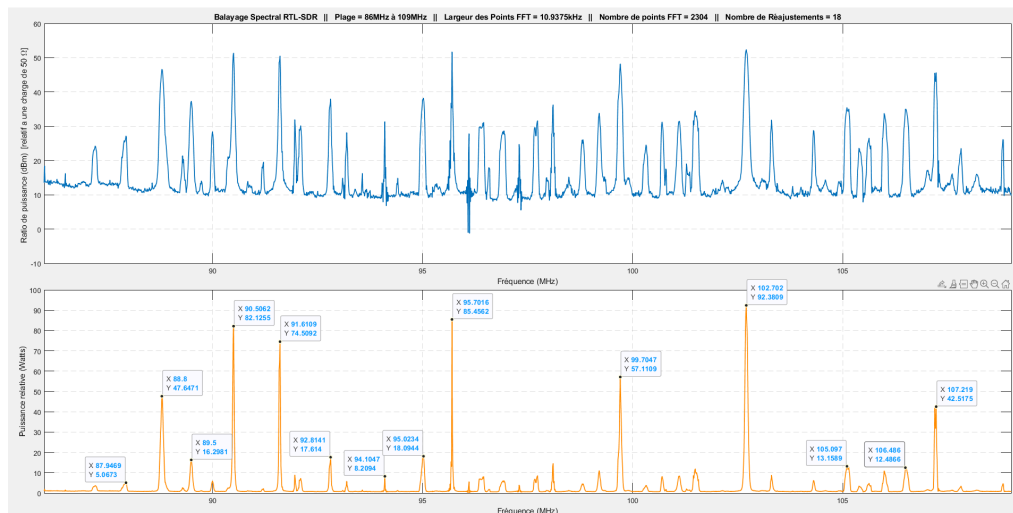
La fréquence d'échantillonnage est définie par la variable `rtlsdr_fs` et est configurée à 2,8 MHz (2 800 000 Hz) :

`rtlsdr_fs = 2.8e6; % RTL-SDR sampling rate in Hz`

XXX - Sur les courbes obtenues, faire un découpage graphique pour identifier les bandes de fréquence des signaux captés pour les zones qui sortent clairement du bruit (voir sur l'échelle linéaire relative). Aidez-vous de la frise des émissions FM radiofréquences de l'ANFR, nommez ces stations.



Si on sort l'antenne dehors on capte bien mieux les fréquences radio, on en a même de nouvelles :



Les fréquences que l'on observe sont bien dans la bande de fréquences de la FM.
 À l'aide de ce lien on a pu déterminer les radios qui appartenaient à ces fréquences :
https://www.csa.fr/maradiofm/liste_ville?recherche=3&commune=CAEN+-+14000

1	Fréquence (MHz)	Nom de la radio
2	86,01	Mouv'
3	87,9	M Radio
4	88,8	RFM Normandie
5	90,5	Radio Classique
6	91,5	France Culture
7	93,1	Radio Phénix
8	94,49	RCF Calvados-Manche
9	95,7	France Musique
10	96,1	Fun Radio
11	97,6	Skyrock
12	99,7	France Inter
13	100,3	Tendance Ouest
14	100,6	Radio Courtoisie
15	102,7	ICI NORMANDIE
16	105,13	RTL
17	106,005	France Info
18	107,2	BFM Business