Wireless Innovation Forum European Conference on Communications Technologies and Software Defined Radio
Brussels – 27-29 June 2012

Software Radio Spectrum Analyzer

Jérôme PARISOT, Emilien LE SUR, Christophe MOY, Daniel LE GUENNEC, Pierre LERAY

SUPELEC/IETR

27 June 2012

SUPELEC - Campus de Rennes
SCEE – Signal, Communications et Electronique Embarquée
IETR – UMR CNRS 6164
Institut d'Electronique et Télécommunications de Rennes

Project goal

- **Student project**
  - implement real radio on a part-time 3 months project
  - evaluate/dimension SDR capabilities for real-time processing
  - not only for communications
- **System**
  - SDR approach
  - USRP N210 from Ettus research
  - Simulink processing environment

Presentation outline

- Power spectral density
- Simulink implementation on N210 platforms
- Windows implementation on N210 platforms
- Conclusion

Presentation outline

- Power spectral density
- Simulink implementation on N210 platforms
- Windows implementation on N210 platforms
- Conclusion

- **Spectrum**
  - continuous $$X(f) = \int_{-\infty}^{+\infty} x(t).e^{-2i\pi ft} dt$$
  - discrete $$\hat{X}(f) = T\sum_{k=1}^{N} x(k).e^{-2i\pi fkT}$$
    - but : amplitude and phase
    - convergence not guaranteed mathematically
- **Power Spectral Density**
  - auto-correlation

for real signals

$$\gamma_x(\tau) = E\{x(t+\tau).x(t)\}$$

Fourrier transform

$$\Gamma_x(f) = \int_{-\infty}^{+\infty} \gamma_x(\tau).e^{-2i\pi f\tau} d\tau$$

Power Spectral Density

- **Power Spectral Density**
  - it can be shown that $\Gamma_x(f) = \lim E\left[\frac{\left|\hat{X}(f)\right|^2}{N.T}\right]$
    - ➔ periodogram
  - ➔ instead: (Schuster - 1898)

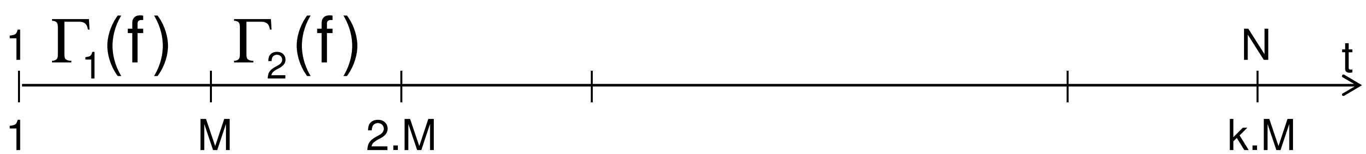
$$\hat{\Gamma}_x(f) = \frac{\left|\hat{X}(f)\right|^2}{N.T}$$

    - but: it can be shown that estimation error standard deviation (hypothesis of White Gaussian Noise)

$$\sigma_{\hat{\Gamma}_x(f_0)} \approx \Gamma_x(f_0)$$

    - error is a the level of the measure!
  - necessary to average
    - Welch approach

- **Discrete PSD by Welch method**
  - based on temporal samples (periodogram-based)
  - subdivide the samples in temporal slots
  - combine the PSD result of each slot in order to make a global mean PSD



$$\hat{\Gamma}_x(f) = \frac{\left|\hat{X}(f)\right|^2}{N.T}$$

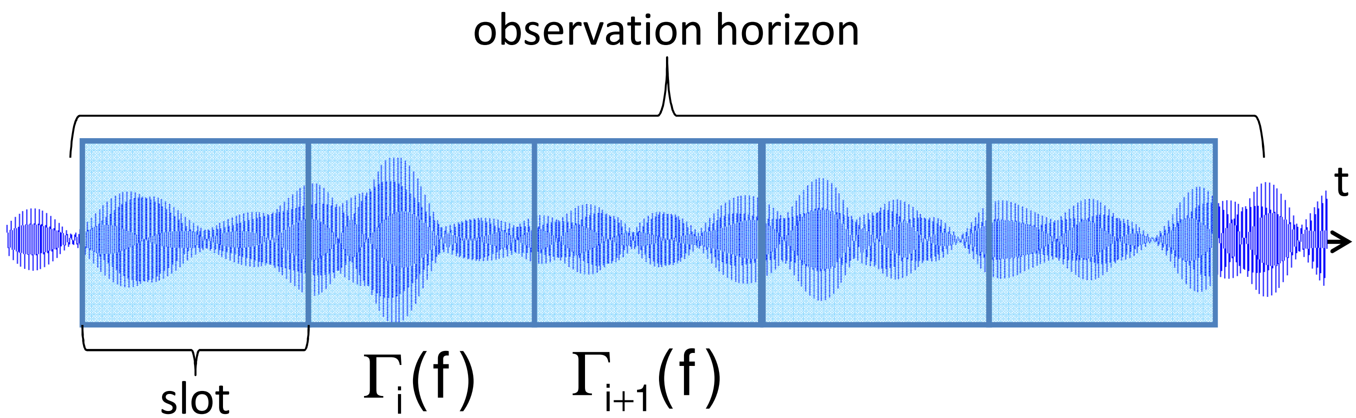
– advantage on precision

$$\sigma_{\hat{\Gamma}_x(f_0)} = \frac{1}{\sqrt{k}}\,\sigma_{\Gamma_i(f_0)}$$

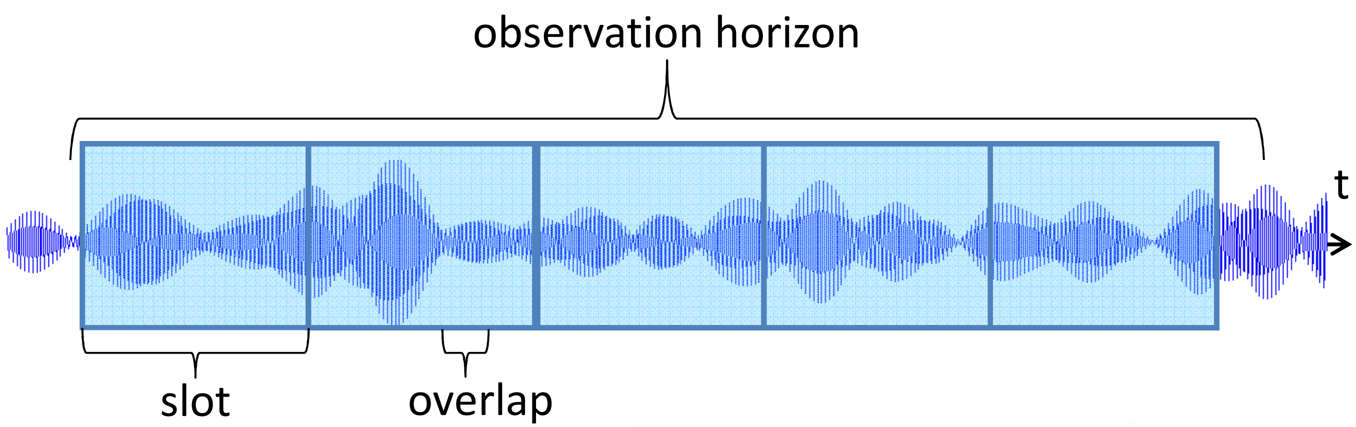
$$\hat{\Gamma}_x(f) = \frac{1}{k}\sum_{i=1}^{k}\hat{\Gamma}_i(f)$$

– disadvantage
  - resolution

WELCH method (II)

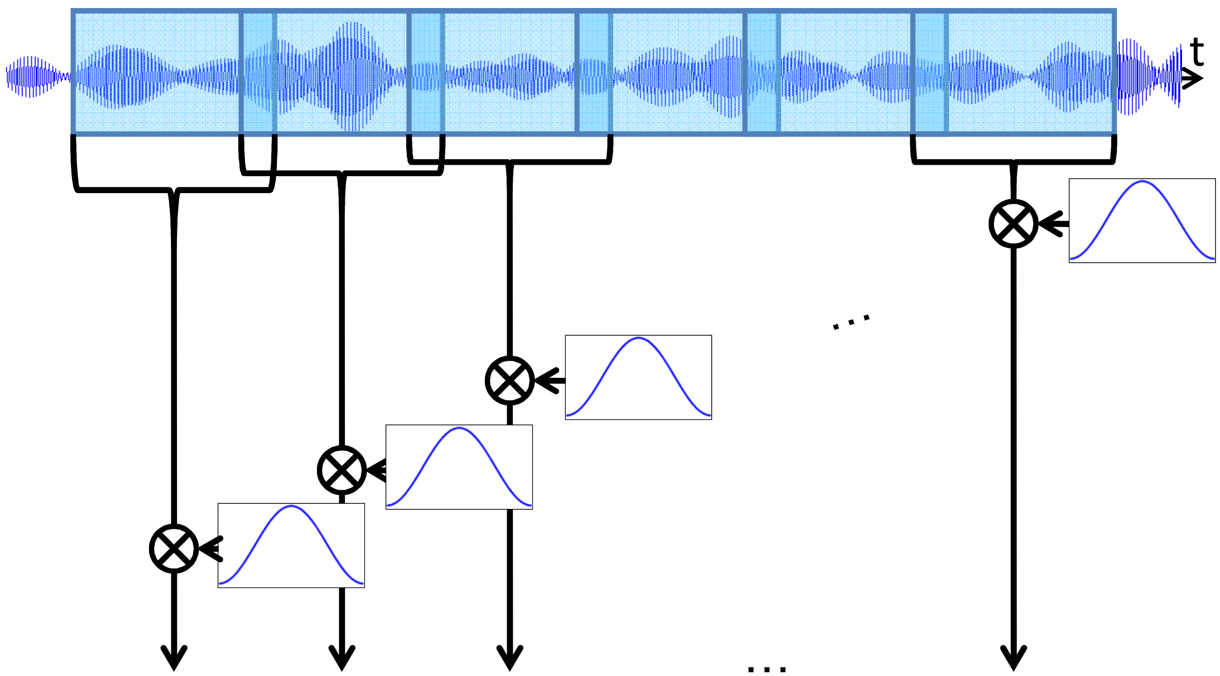


- **Overlap**
  - avoid loosing effects at the slot border

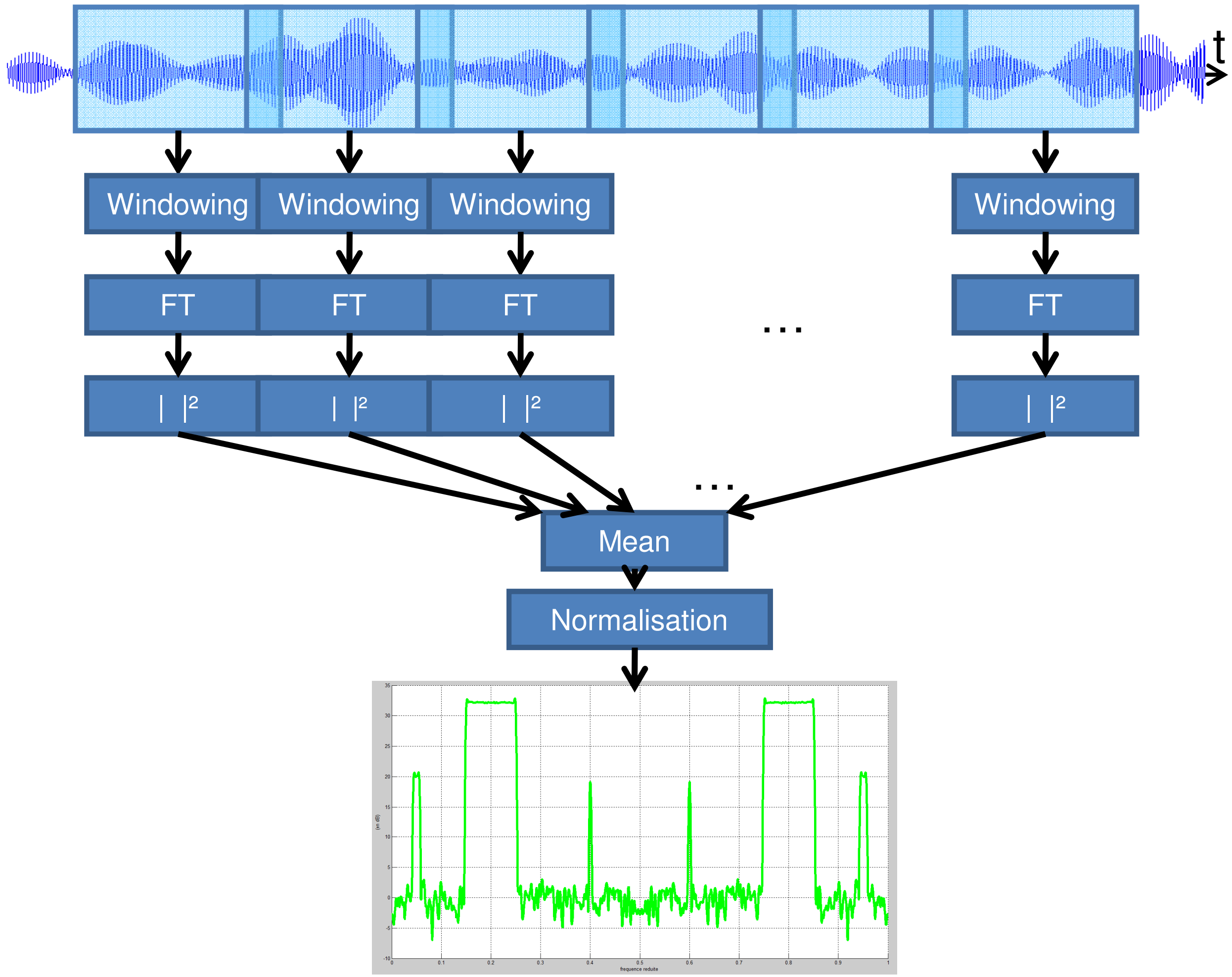


WELCH method (III)

- **Windowing**
  - in order to mitigate truncature effect
  - Hamming here



WELCH method (IV)



Matlab implementation

- **Function for Welch algorithm**
  - input: time samples
  - output: PSD
  - parameters:
    - number of slots
    - overlapping ratio
    - windowing type

```
signalOut = zeros(1,sizeFFT);
 .
for i=0:nbSlots-1

    offset=doffsetSlot*i+1;
    % Slot extraction and windowing.
    slot= signalIn(offset:offset + slotSize-1).*fen;
    % Normailze FFT.
    S = fft(slot, sizeFFT)/slotSize;
    % Square
    signalOut = signalOut + abs(S).^2;

end

% Mean and normalization.
signalOut = signalOut/ nbSlots * slotSize/ norm_win;
```

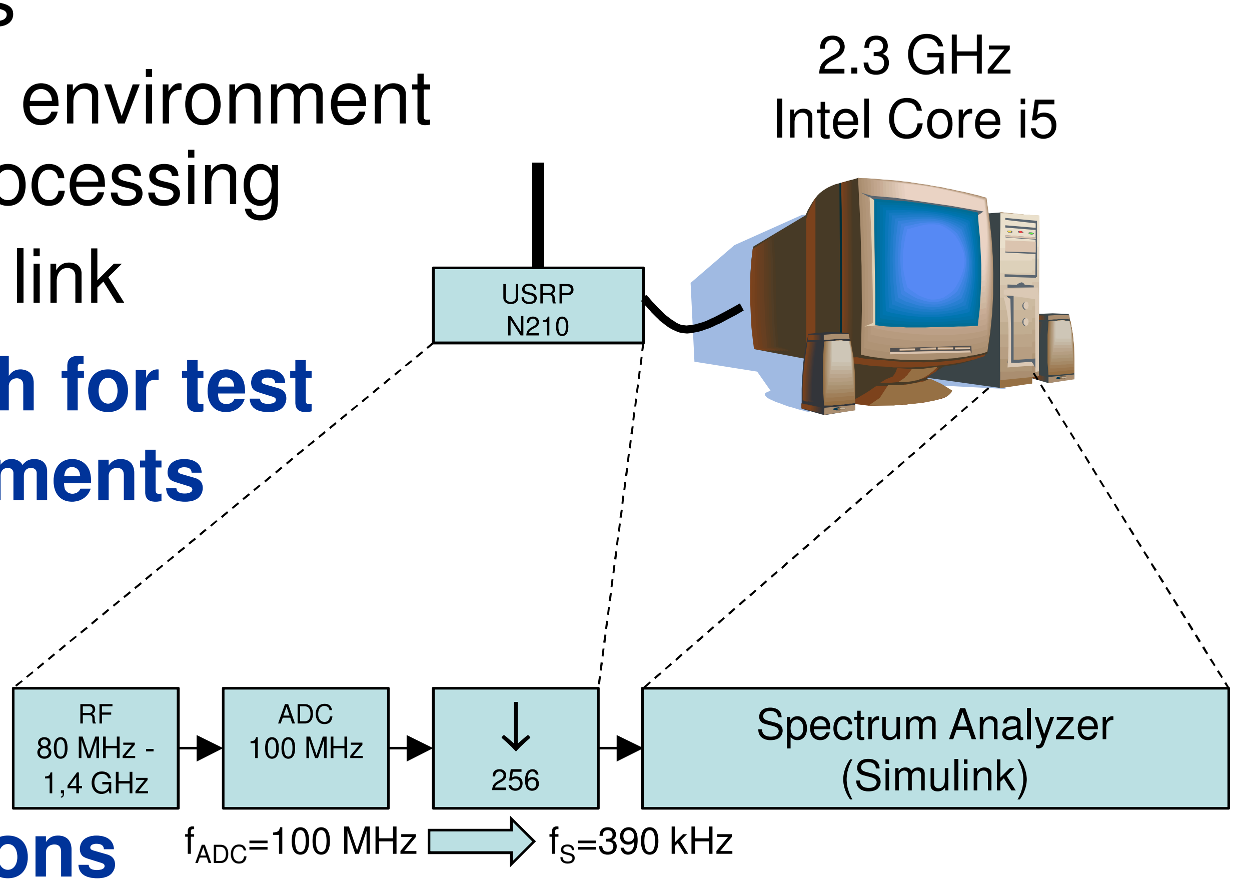
```
function [ signalOut ] = algoWelch( signalIn , nbSlots , overlapRatio, window)
```

Presentation outline

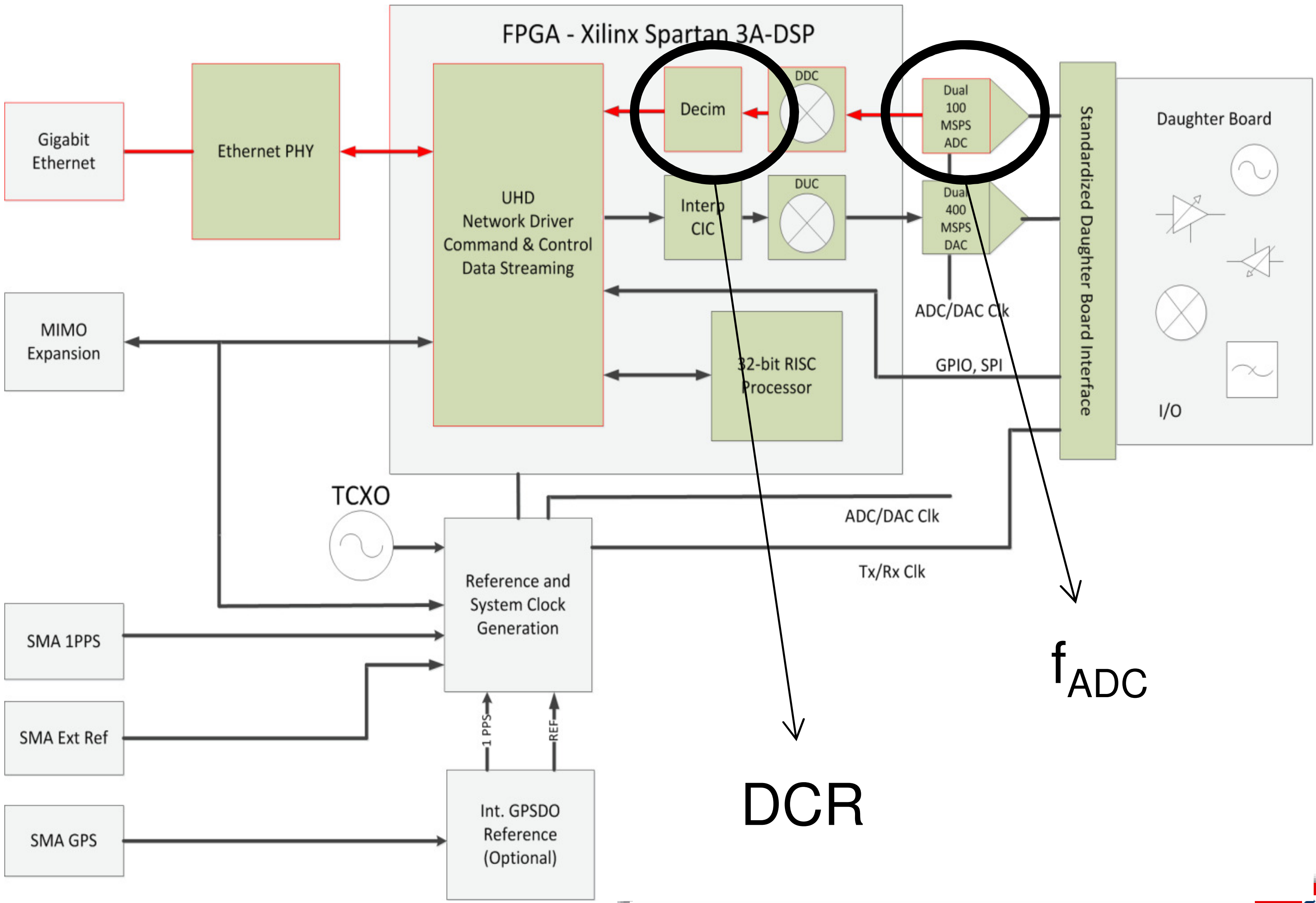
- Power spectral density
- **Simulink implementation on N210 platforms**
- Windows implementation on N210 platforms
- Conclusion

Hardware system

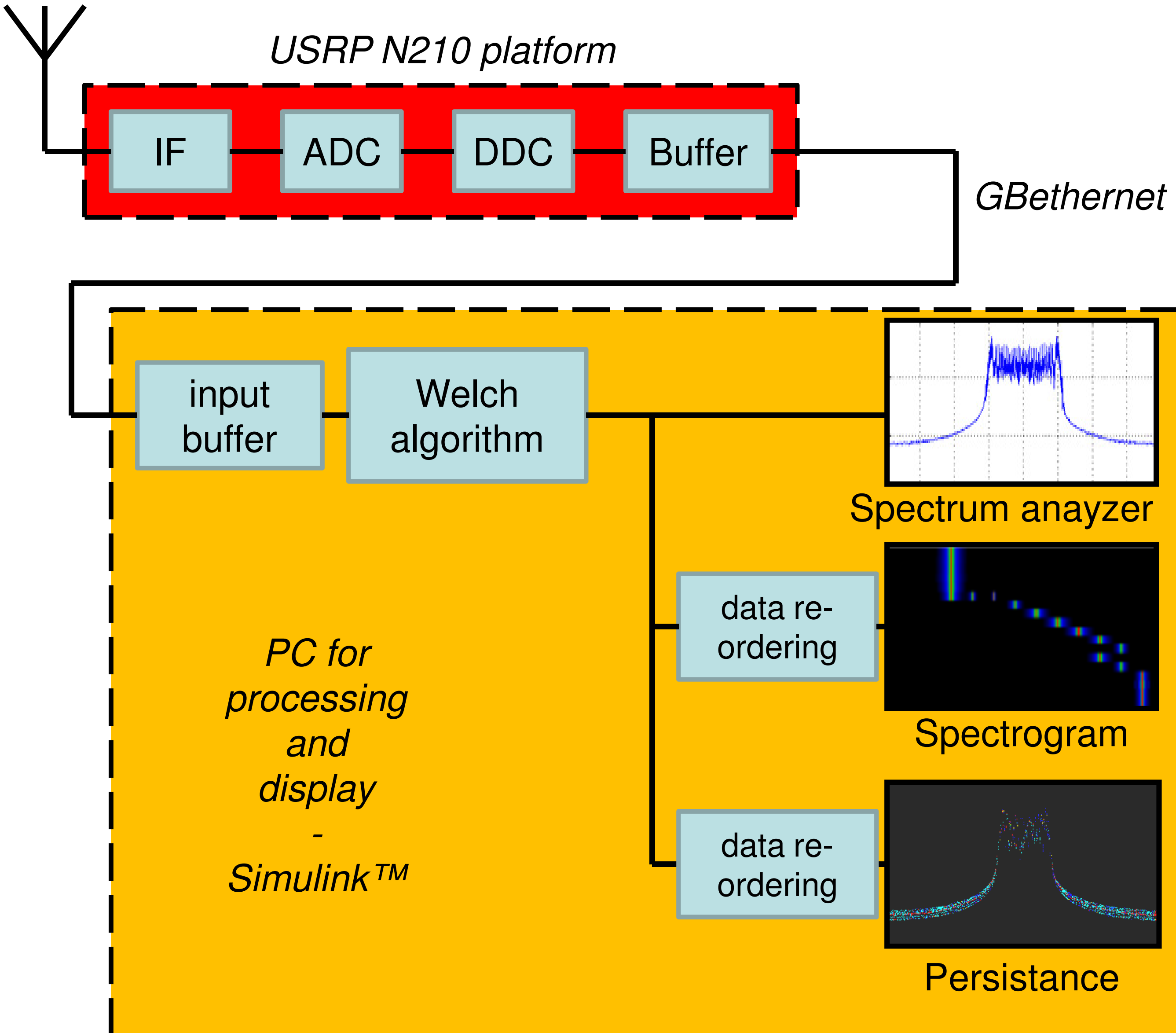
- **USRP based platform (N210 frome Ettus™)**
  - UHD drivers
  - for Simulink environment real-time processing
  - GBethernet link
- **SDR approach for test and measurements**
- **Not only for communications**



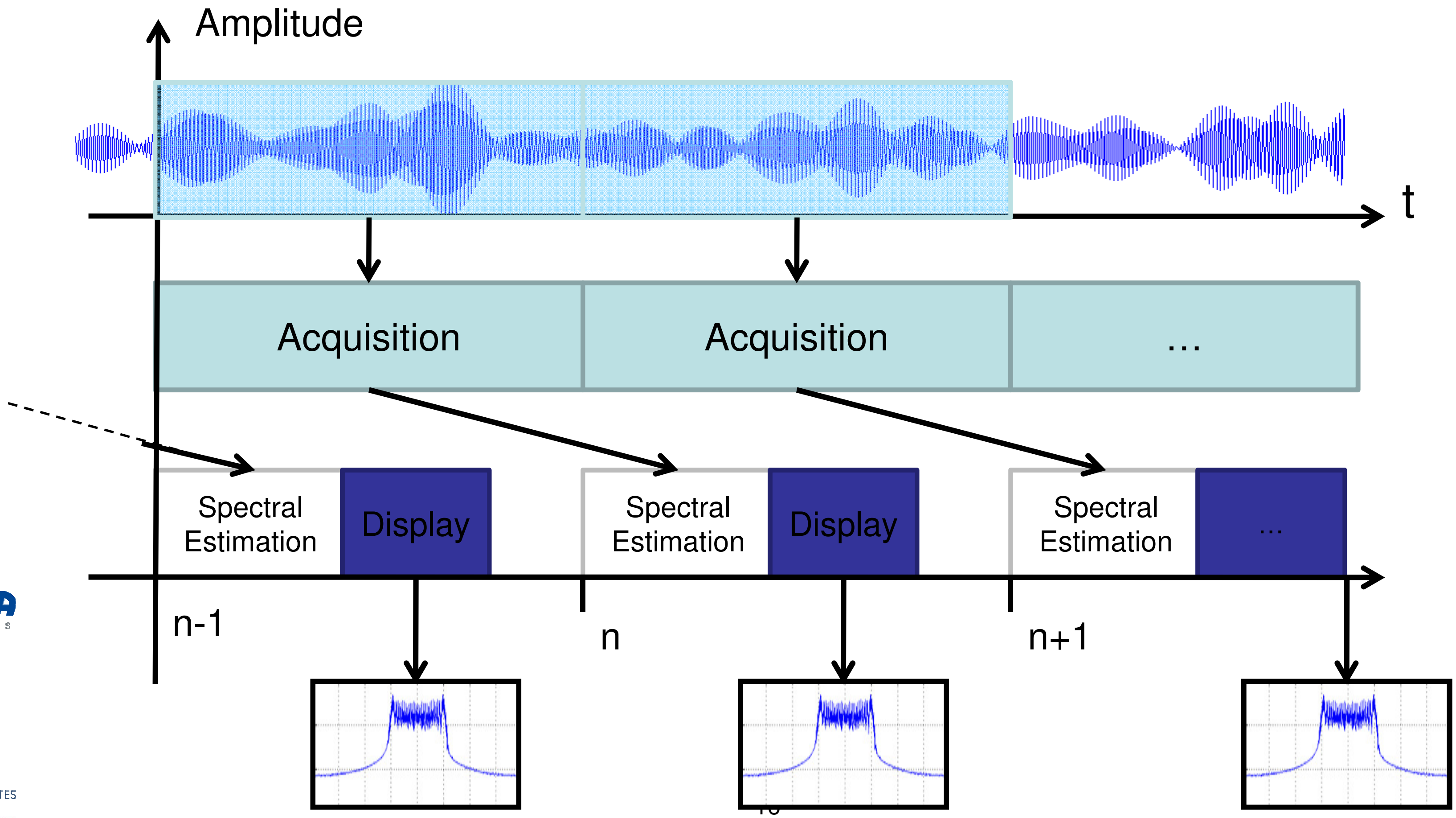
N210 platform from Ettus™



System functional view



- Spectrum analyzer



Simulink processing (I)

- **Spectrum analyzer performance**
  - bandwidth
    - DCR: undersampling factor

$$f_{Max} = \frac{f_{ADC}}{2 \cdot DCR}$$

  - display frequency (of PSD)
    - n_f: number of samples per frame
    - n_b: number of frames

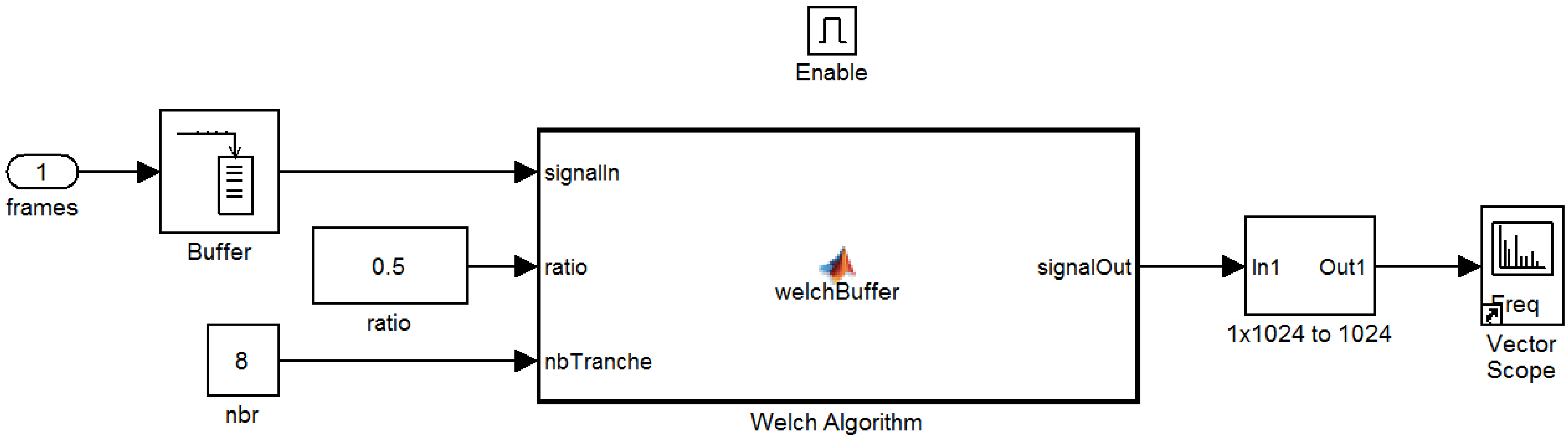
$$f_{disp} = \frac{f_{ADC}}{DCR} \cdot \frac{1}{n_f \cdot n_b}$$

  - real time (2.3 GHz Intel Core i5)

Fc (center freq.)	Decim	bandwidth	Frame Length	Sample Time (s)	Output data type
80 Mhz – 1.4 GHz	256	195 kHz	362	$2{,}56.10^{-6}$	double

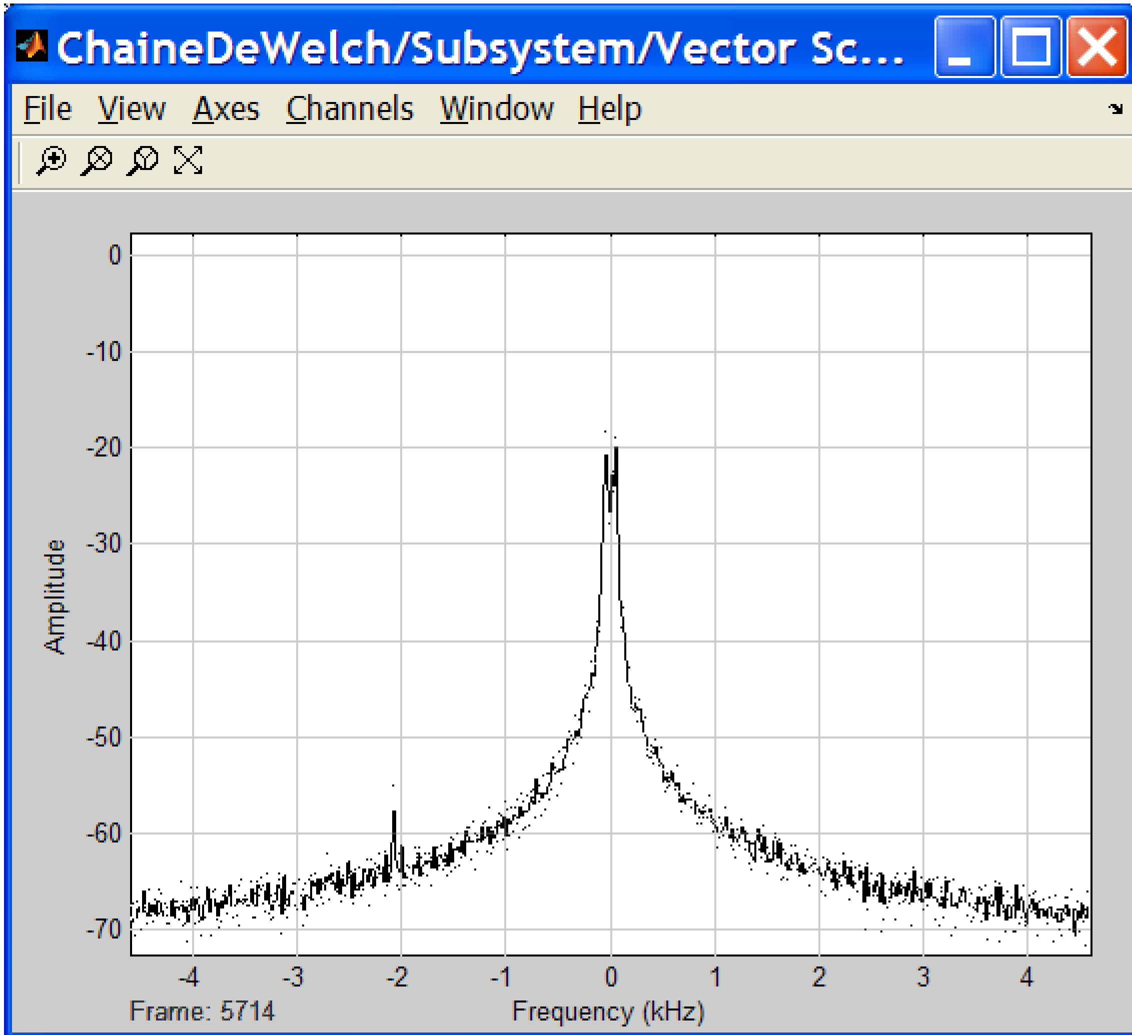
Simulink processing (I)

- Spectrum analyzer in Simulink

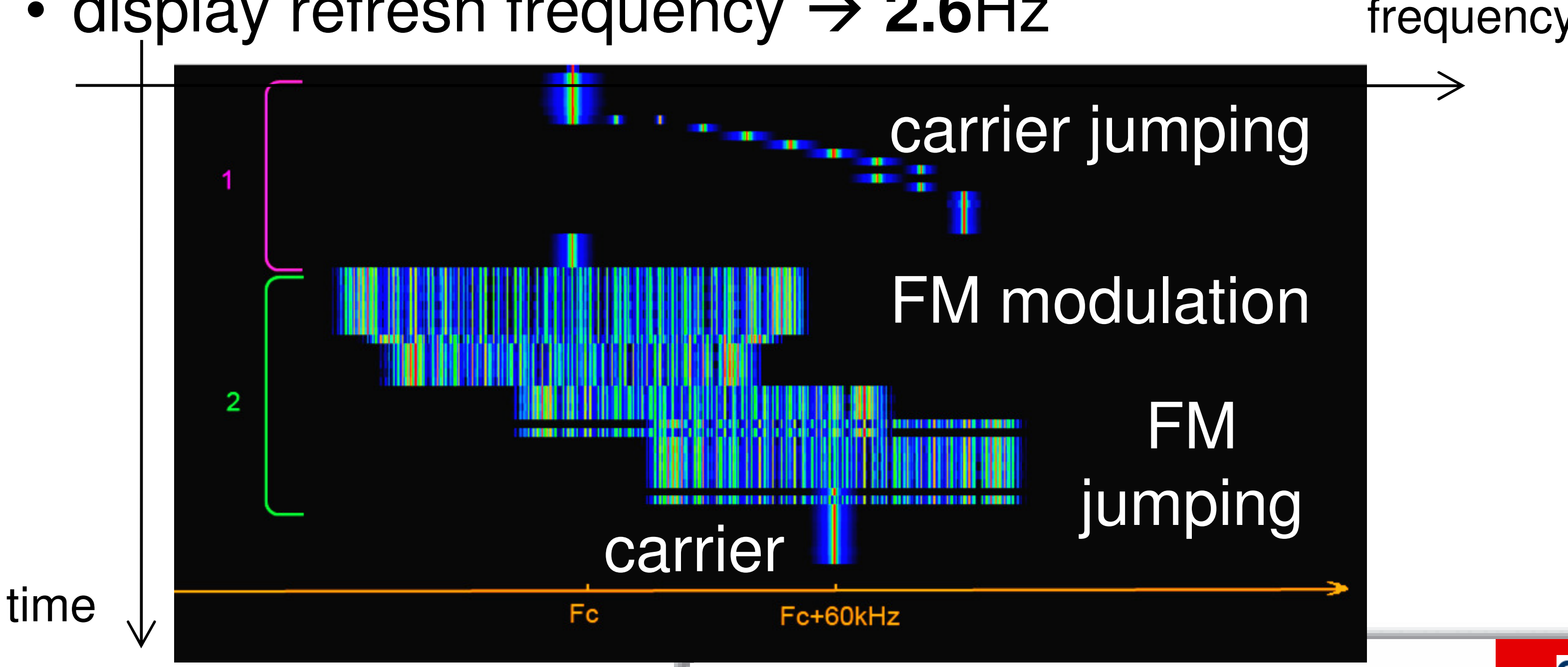


Simulink processing (I)

- **Spectrum analyzer**
  - FM 10 kHz signal
  - DCR = 256
  - DCR = 128
  - DCR = 64

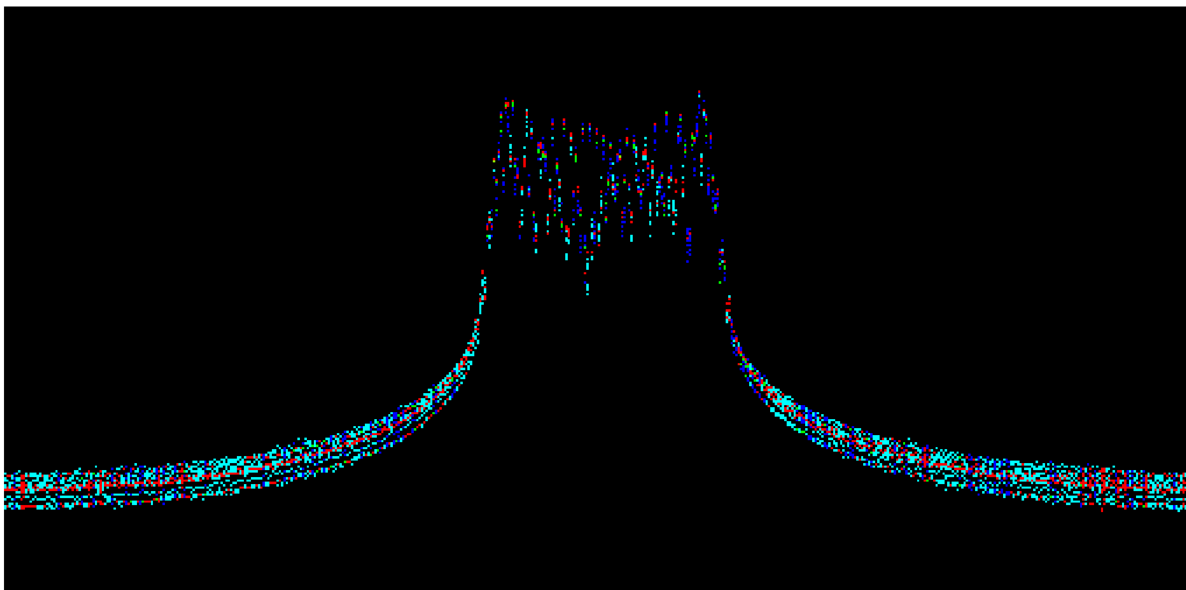


- **Spectrogram**
  - overlapping ratio = 0.5, Nb slots = 8
    - display refresh frequency → **5.7**Hz
  - overlapping ratio = 0.8, Nb slots = 16
    - display refresh frequency → **2.6**Hz



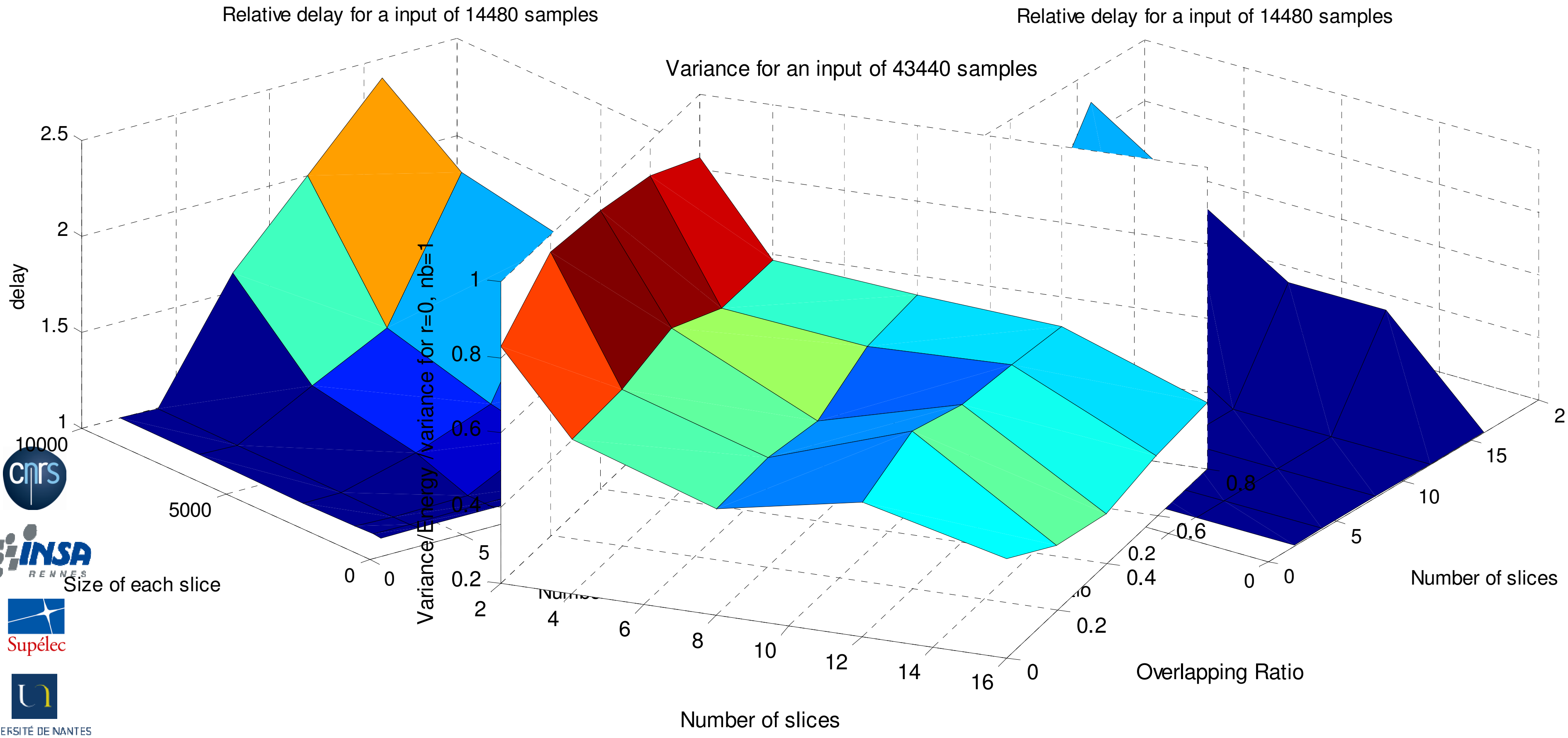
Simulink processing (III)

- **Remanence**
  - overlapping ratio = 0.5, Nb slots = 8
    - display refresh frequency → **8.33** Hz
  - overlapping ratio = 0.8, Nb slots = 16
    - display refresh frequency → **3.86** Hz



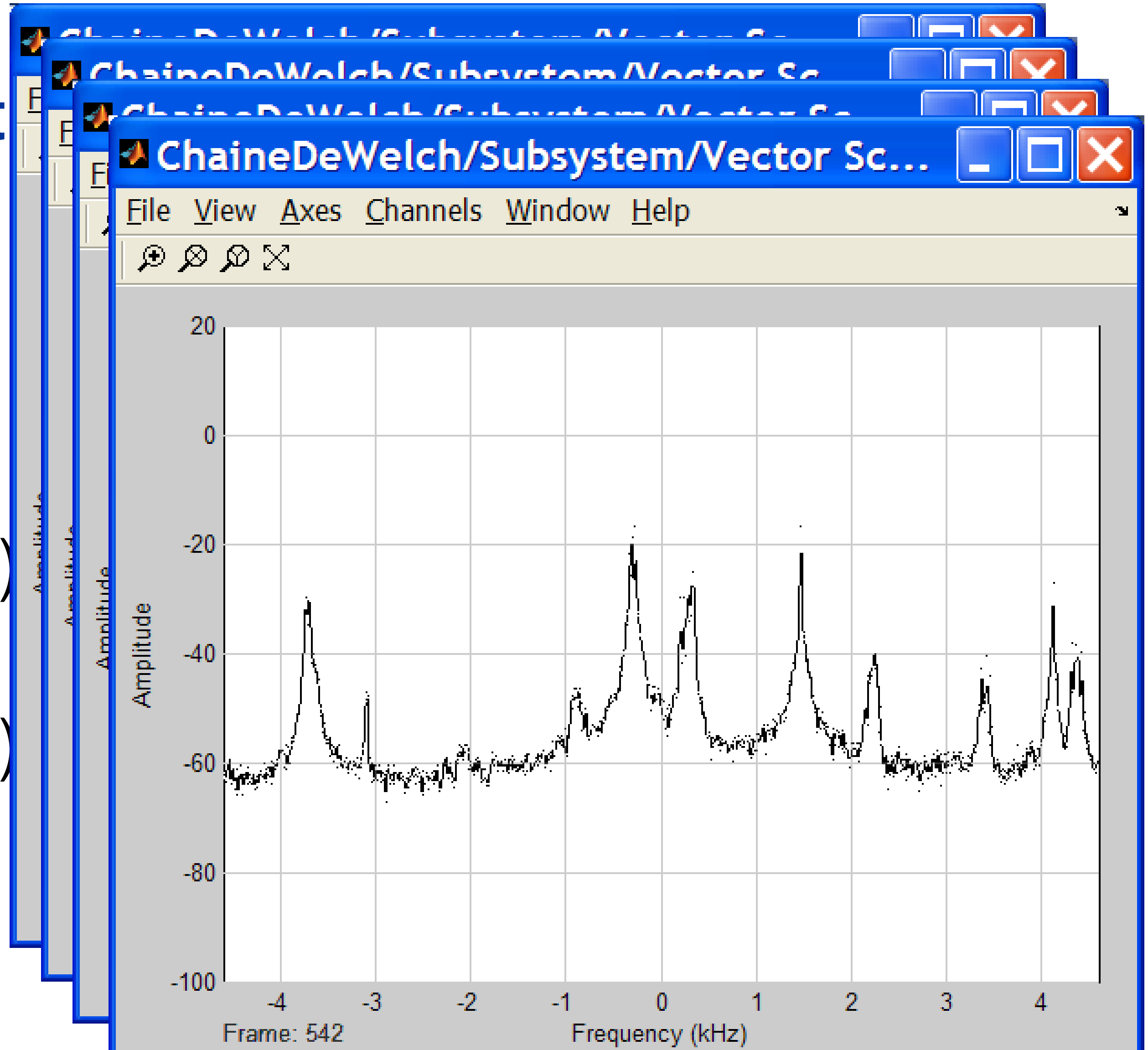
PSD variance estimate

- **Trade-off: quality / execution duration**



screen shots

- **FM broadcast**
  - 256
  - 512
  - 64 (delayed)
  - 16 (delayed)



Presentation outline

- Power spectral density
- Simulink implementation on N210 platforms
- **Windows implementation on N210 platforms**
- Conclusion

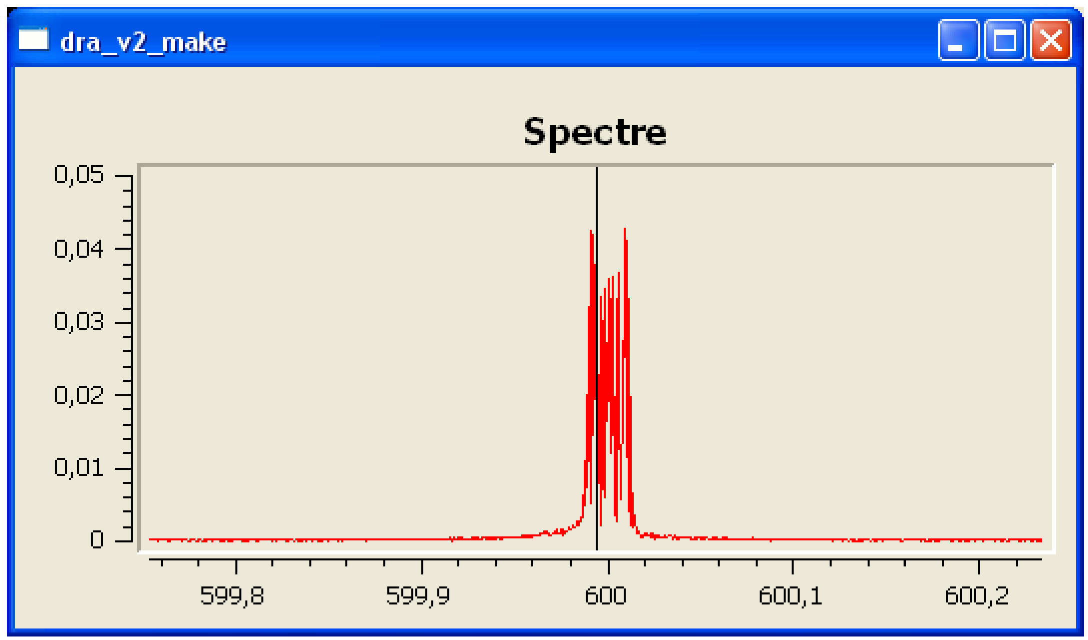
Windows implementation

- **Development Environment**
  - SUPELEC proprietary environment (Windows)
- **UHD library**
  - from Ettus Research
- **Supporting HDCRAM**
  - **H**ierarchical and **D**istributed **C**ognitive **R**adio **A**rchitecture **M**anagement [1]
  - HDCRAM is an architecture for the management of reconfiguration and cognitive facilities (metrics capture and decision/learning)
  - for real-time auto-adaptation

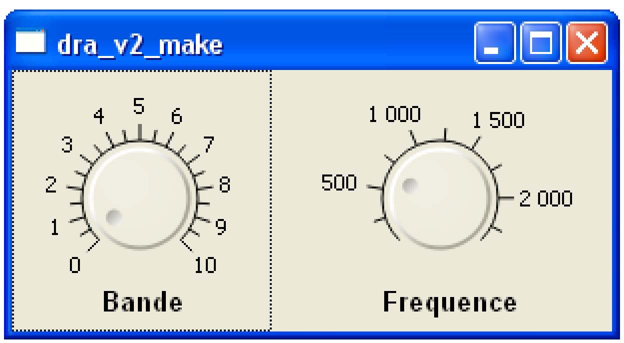
[1] Christophe MOY, "High-Level Design Approach for the Specification of Cognitive Radio Equipments Management APIs", Journal of Network and System Management - Special Issue on Management Functionalities for Cognitive Wireless Networks and Systems, vol. 18, number 1, pp. 64-96, Mar. 2010

HDCRAM management

- **Spectrum analyzer is a kind of cognitive radio**
  - carrier frequency adjustment ➔ sensor
  - DCR factor (undersampling) ➔ sensor
  - display parameters ➔ data processing to be reconfigured



- **FM 10 kHz**



- Power spectral density
- Simulink implementation on N210 platforms
- Windows implementation on N210 platforms
- **Conclusion**

Conclusion

- **Teaching level (project benefit for students)**
  - power spectrum density study
  - fast and easy to implement
  - very motivating for students compared to
    - analytical analysis
    - simulations more or less deconnected from reality
    - writing a paper for a conference
- **SDR for other stuff than communications**
  - channel sounder last year
  - spectrum analyzer here
- **Sensor for cognitive radio**

- **Thank you to/This is the work of students:**

Emilien LE SUR

Jérôme PARISOT

Thanks for your attention

http://www.rennes.supelec.fr/ren/perso/cmoy/SCEE-SERI/