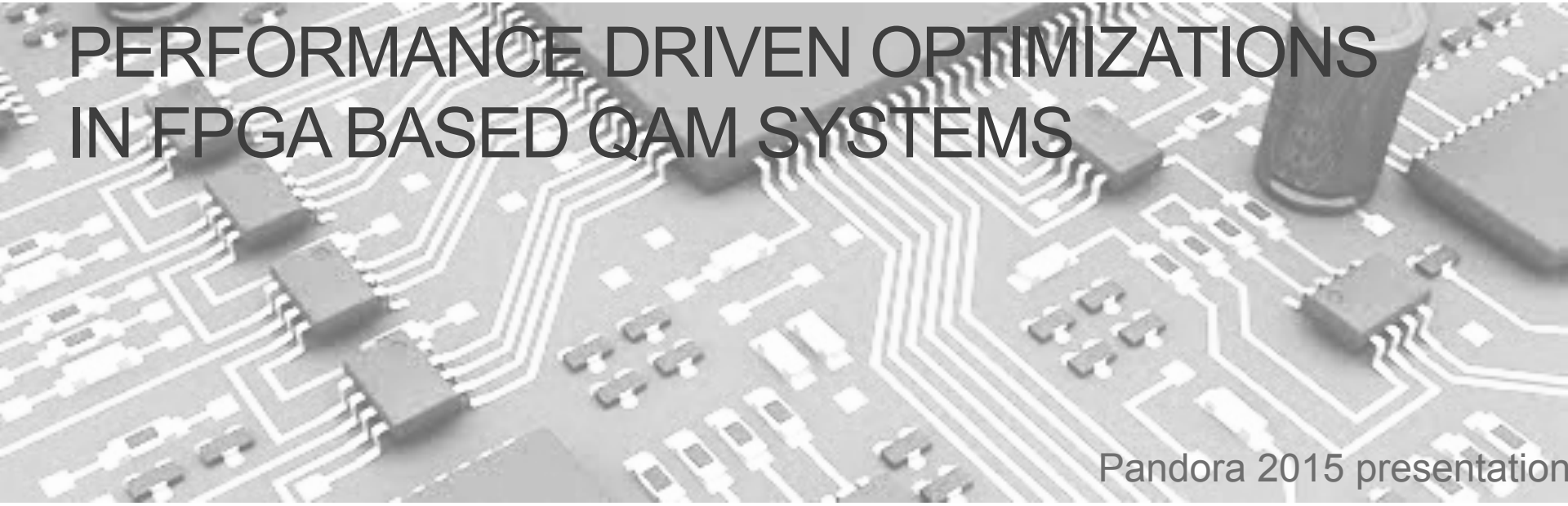
Master Thesis ITIV, Embedded Systems Group
Karlsruhe Institute of Technologies



PERFORMANCE DRIVEN OPTIMIZATIONS IN FPGA BASED QAM SYSTEMS

Pandora 2015 presentation

Alberto Sonnino
(1794121)

Introduction & Motivation

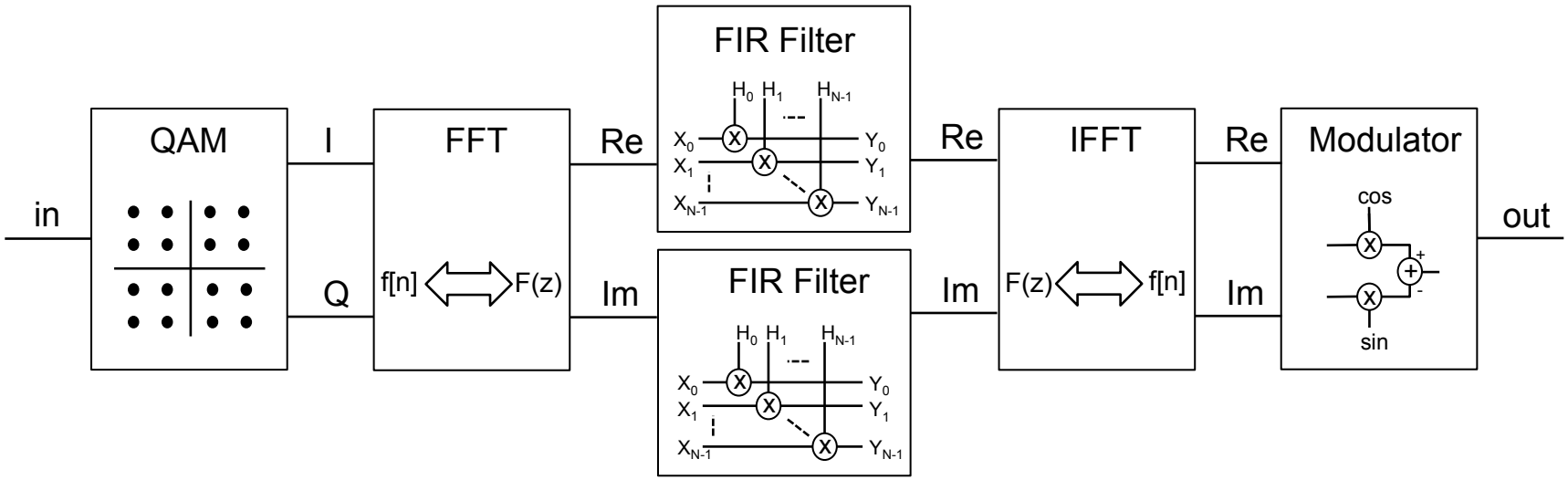
- Context:
  - High SNR (Signal to Noise Ration) and high data rate requires advanced techniques.
  - Contribution to reach terabit's communication in future application.

- Hardware Choice:
  - The great configurability, flexibility and low cost make FPGAs an adequate hardware support for this project.
  - Currently clocked below 1GHz: need improvement on all abstraction layers.
  - Implementation based on Xilinx Virtex 7 FPGA kit.

- Modulation Choice:
  - Requirement of advanced modulation able to carry many bits of information per symbol. Quadrature Amplitude Modulation (QAM) is widely used in digital communications.
  - Higher order QAM implies higher data rate but may cause symbol's misinterpretation at the receiver, therefore, 16-QAM is chosen.

- Filter Choice:
  - Needed to avoid ISI (Inter Symbols Interferences) and noise by limiting the transmission band.
  - FIR (Finite Impulse Response) filters have linear phase, same error per cycle (no feedback) and inherent stability. Squared Root Raised Cosine (SRRC) filter is selected.
  - No filter parameter's optimization but efficient implementation.

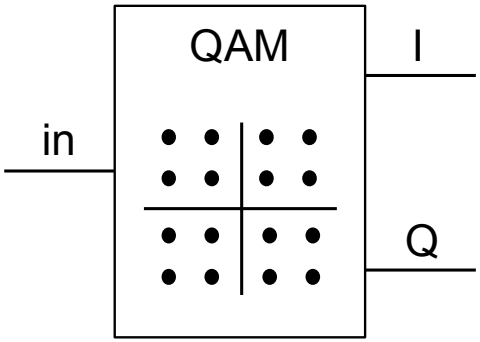
- On this paper:
  - Performance optimization of QAM transmitter, exploiting the degree of parallelism of the FPGA platform and mixed-domains (time and frequency) where beneficial.
  - Start with a general view of the transmitter and then goes deeper and deeper in the abstraction layers.

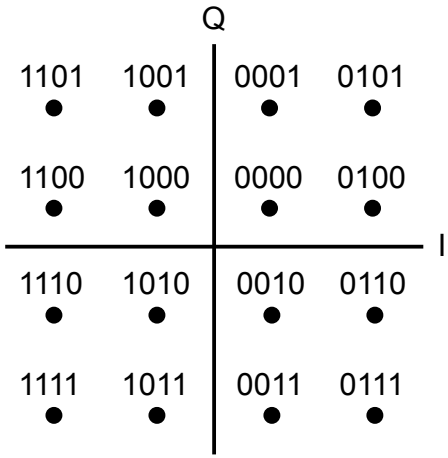


Fundamentals

- QAM mapping:
  - QAM is efficiently represented by a constellation diagram, standard rectangular constellation is preferred in this work for his poor overhead and simplicity.
  - Multiples QAM order format are possible (16-QAM, 32-QAM, 64-QAM, etc.), denoted by M-QAM.
  - M denotes the number of points in the constellation, each symbol contains $\log_2(M)$ bits.
  - Higher order QAM format implies higher data rate but may causes symbol's misinterpretation at the receiver, therefore, 16-QAM is chosen.
  - Gray Code associates a symbol to a given constellation position ensuring one bit difference between adjacent symbols.
  - 16-QAM clusters the input by groups of 4 bits and apply:

$$\left[\underbrace{b_3}_{I_1} \; \underbrace{b_2}_{Q_1} \; \underbrace{b_1}_{I_2} \; \underbrace{b_0}_{I_2} \right] \quad \Rightarrow \quad I = \{I_1, I_2\} \quad \text{and} \quad \mathrm{Q} = \{\mathrm{Q}_1, \mathrm{Q}_2\}$$





- # Fourier Transform:
  - Fourier Transform decomposes the signal in an alternative representation made of sinus and cosinus.
  - For discrete inputs, the Discrete Fourier Transform (DFT) sends a signal from the time to the frequency domain and, the Inverse Discrete Fourier Transform (IDFT) takes it back:

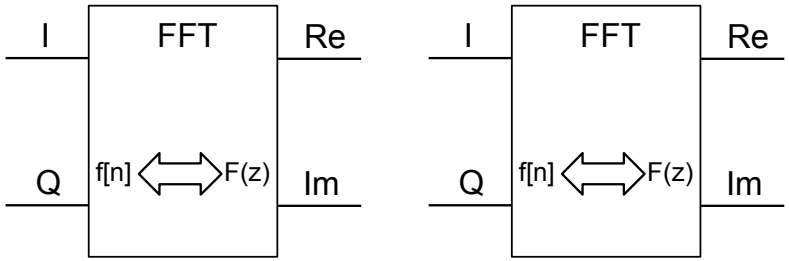
$$X[k] = \sum_0^{N-1} x[n] e^{-2\pi i k n / N} \qquad k \in \mathcal{Z}$$

$$x[n] = \frac{1}{N} \sum_0^{N-1} X[k] e^{2\pi i k n / N} \qquad n \in \mathcal{Z}$$

  - Linear operations in time domain have an equivalent in frequency domain: convolution in time domain becomes a simple multiplication in frequency domain.

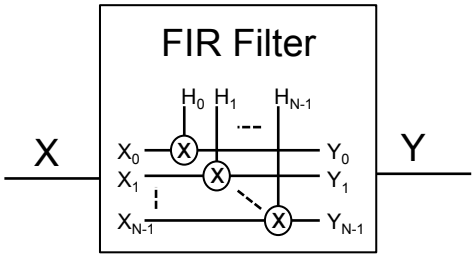
$$\begin{aligned} \mathcal{F}\{f * g\} &= \mathcal{F}\{f\} \cdot \mathcal{F}\{g\} \\ &= G \cdot F \end{aligned} \qquad \begin{aligned} f * g &= \mathcal{F}^{-1}\{\mathcal{F}\{f\} \cdot \mathcal{F}\{g\}\} \\ &= \mathcal{F}^{-1}\{G \cdot F\} \end{aligned}$$

  - Fast Fourier Transform (FFT) is an efficient algorithm to compute DFT. Computational complexity of DFT is $O(N^2)$ while FFT has a complexity of $O(N\log_2(N))$.



  - The inverse transform can be computed using the Inverse Fast Fourier Transform (IFFT).

- Filter:
  - Limit the transmitted signal into a defined part of the channel to prevent interferences with adjacent channels: a filter respecting Nyquist criteria avoids ISI.
  - FIR filters posses a linear phase, a same error every cycle (no feedback) and inherent stability.

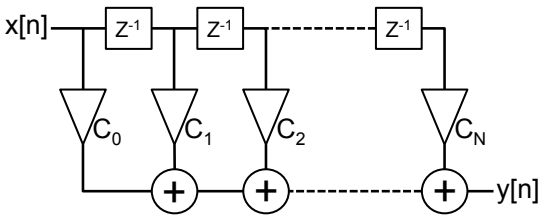


  - Matched filters optimize the SNR (if only stochastic noise) by convolving the input with an expected template of this input.
  - A filter having all those properties is the SRRC filter and constitute a good compromise between spectral efficiency and low ISI.

Time Domain

$$y[n] = x[n] * h[n]$$

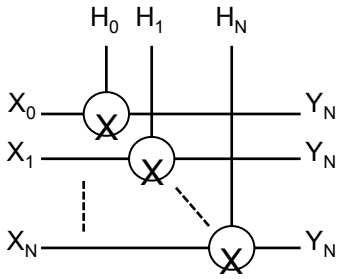
- Convolution: difficultly parallelizable



Frequency Domain

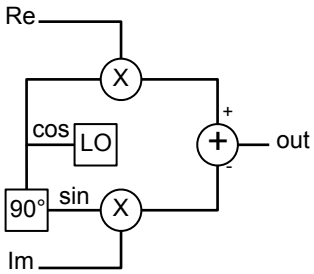
$$Y[k] = X[k] \cdot H[k]$$

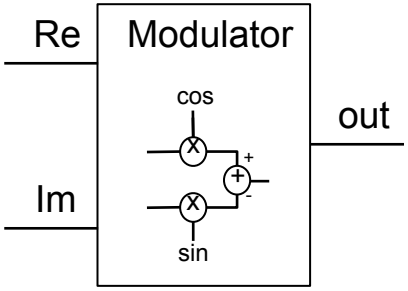
- Multiplication: easily parallelizable



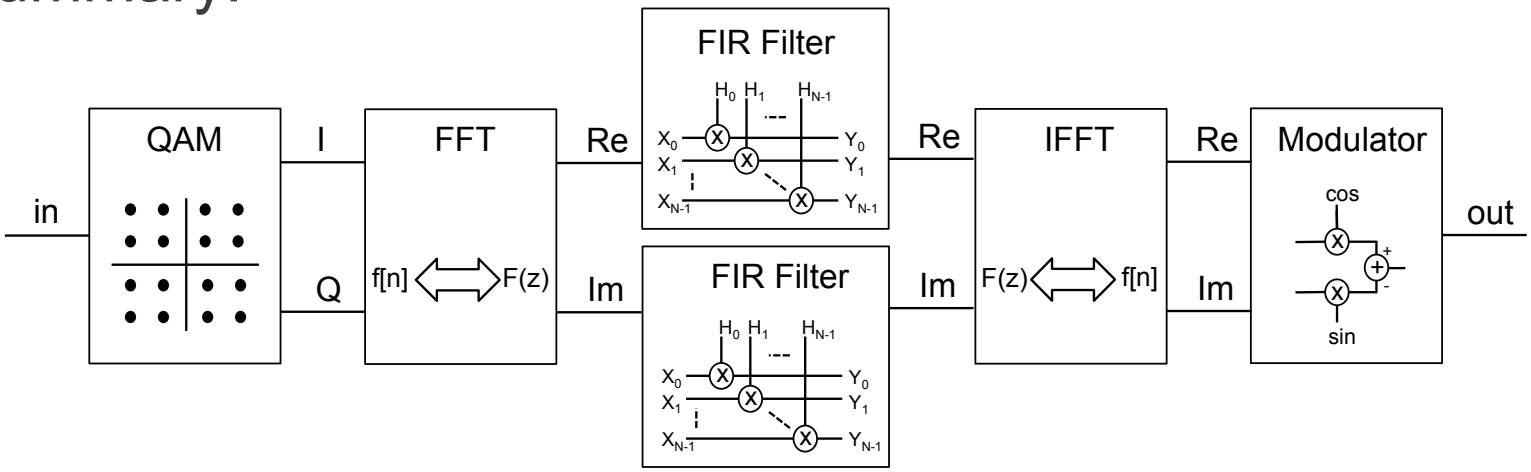
- Modulator:
  - Local Oscillator (LO) generate the carrier sinusoidal waves at a fixed frequency f_0. I component is multiplied by cos and Q component by sin (or cos shifted by 90°).
  - Then, we subtract those two products and the transmitter outputs the result.

$$\begin{aligned} out(t) &= \mathcal{R}\left\{[I(t) + iQ(t)]e^{2\pi f_0 t}\right\} \\ &= I(t)\cos(2\pi f_0 t) - Q(t)\sin(2\pi f_0 t) \end{aligned}$$





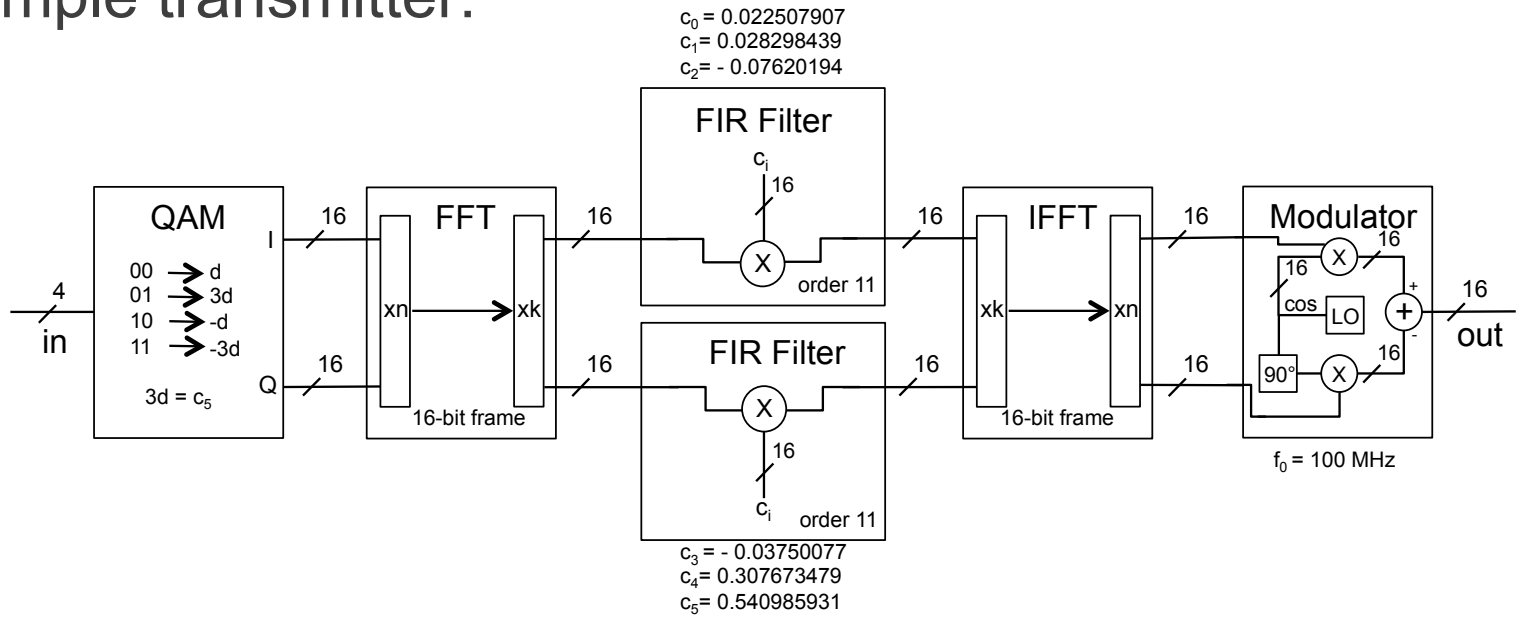
- Summary:



Concepts & Methodology

- Strategy:
  1. Build a single channel (without parrallelization) mixed-domain modulator.
  2. Compare the output of each block with a MATLAB simulation (considered as perfect).
  3. Optimize this single channel tranmitter to reach high frequencies and low area usage.
  4. Parrallelize the transmitter.

- Simple transmitter:



Summary & Further Improvements

- Topic:
  - Performance optimization of QAM transmitter, exploiting the degree of parallelism of the FPGA platform and mixed domains (time and frequency) where beneficial.

- Some Results:

Clock	**62.5 MHz**
Parallel inputs	**16**
DSP	**57%**
LUTs	**49%**
Effective speed	**16 * 62.5 = 1GHz**

- Further Improvements:
  1. Try implementation with DSP48E1 logic core.
  2. Optimize parallelization.

Thanks for your attention !