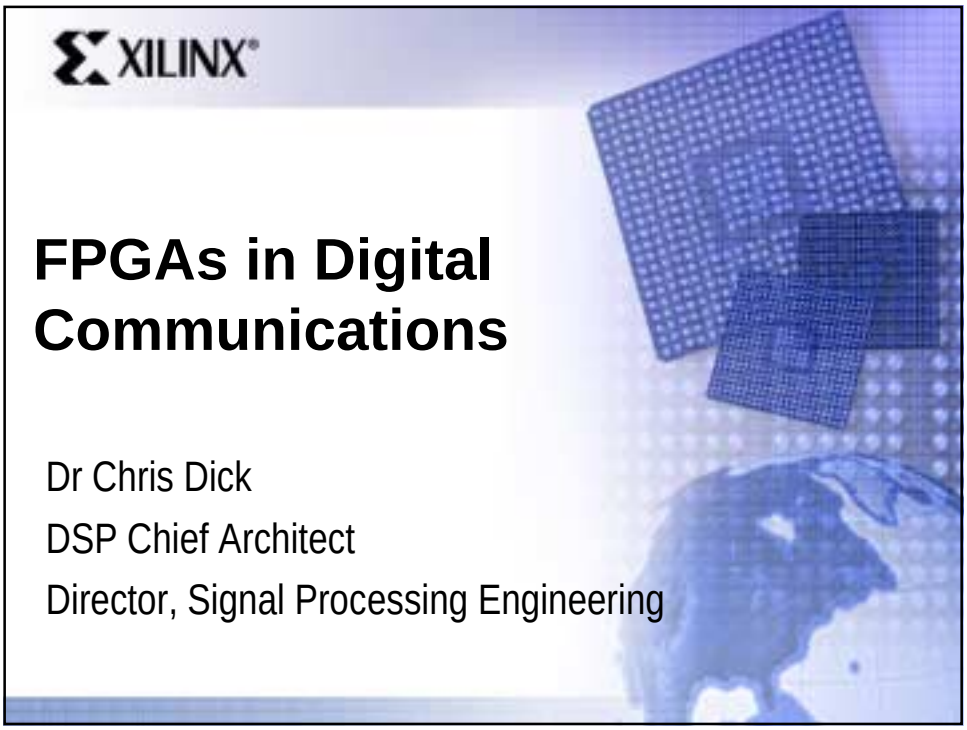
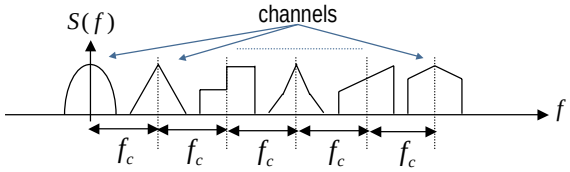


Digital Comm Examples

- Channelized receiver to support
  - QAM
  - OFDM
- Examine
  - Channelizer implementation
  - QAM demodulator architecture
  - OFDM modulator/demodulator

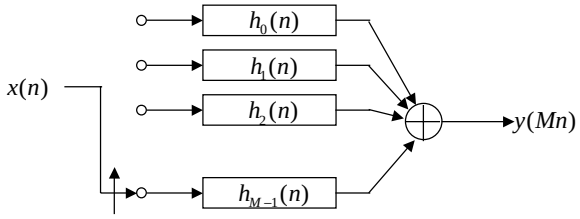
Passband Polyphase Filters



- In a FDM digital communication system a common requirement is, for each channel:
  - translate the channel to baseband
  - shape the channel spectrum
  - reduce the sample rate to match the channel bandwidth
- This is the function of a *channelizer*
- When the channel spacing is equal a computationally efficient structure for performing the above functions is the carrier centered polyphase transform

Channelization 3

Baseband Polyphase Filter



$h_0(n) =$	h_0	h_M	$\cdots$	h_{N-M}
$h_1(n) =$	h_1	h_{M+1}	$\cdots$	h_{N-M+1}
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
$h_{M-1}(n) =$	h_{M-1}	h_{2M-1}	$\cdots$	h_{N-1}

Channelization 4

Passband Polyphase Filters

Express the filter coefficient set in terms of a course and vernier index r_1 and r_2 respectively

$$h(n) = h(r_1 + Mr_2) \qquad r_1 = 0, \ldots, M-1, \quad r_2 = 0, \ldots, \frac{N}{M} - 1$$

Invoke the modulation theorem to convert a prototype baseband filter to its equivalent carrier centered, or spectrally shifted version

if $h(n) \Leftrightarrow H(\theta)$

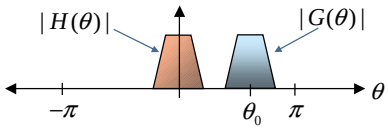
then $h(n)e^{j\theta_0 n} \Leftrightarrow H(\theta - \theta_0)$

Channelization 5

Passband Polyphase Filters

The coefficients of the carrier centered filter are

$$g(n) = h(n)e^{j\theta_0 n}$$



Now perform a polyphase partition on the modulated coefficients

$$g_{r_1}(r_2) = h(r_1 + Mr_2)e^{j\theta_0(r_1 + Mr_2)}$$
$$= h(r_1 + Mr_2)e^{j\theta_0 r_1}e^{j\theta_0 Mr_2}$$

Select θ_0 so that a single period of the series $e^{j\theta_0 n}$ is harmonically related to M

Channelization 6

Passband Polyphase Filters

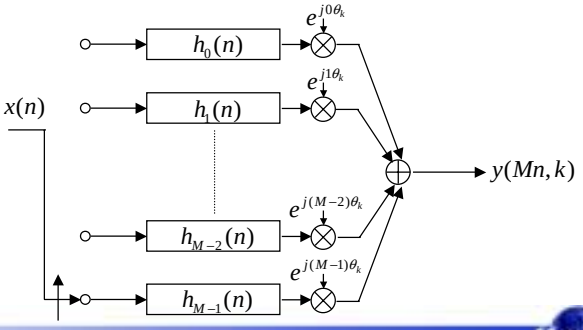
$$\theta_0 = k\frac{2\pi}{M}$$

$$g_{r_1}(r_2) = h(r_1 + Mr_2)e^{j\theta_0 r_1}e^{jk\frac{2\pi}{M}Mr_2}$$

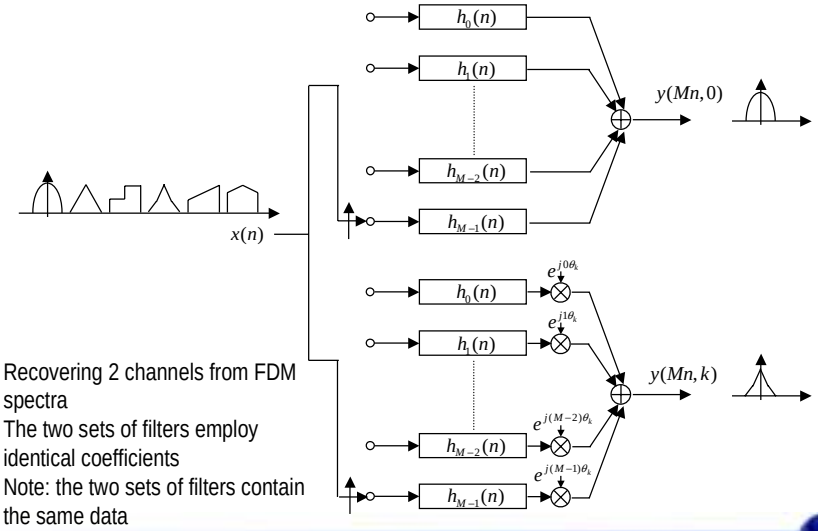
$$= h(r_1 + Mr_2)e^{jk\frac{2\pi}{M}r_1}$$

Carrier centered polyphase filter the one structure

- Translates the channel to baseband
- Shapes the signal
- Reduces the sample rate



Passband Polyphase Filters



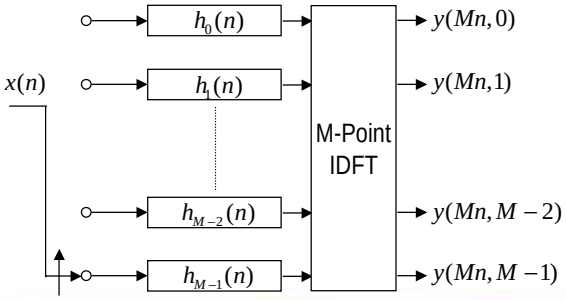
Recovering 2 channels from FDM spectra
The two sets of filters employ identical coefficients
Note: the two sets of filters contain the same data

Polyphase Transform

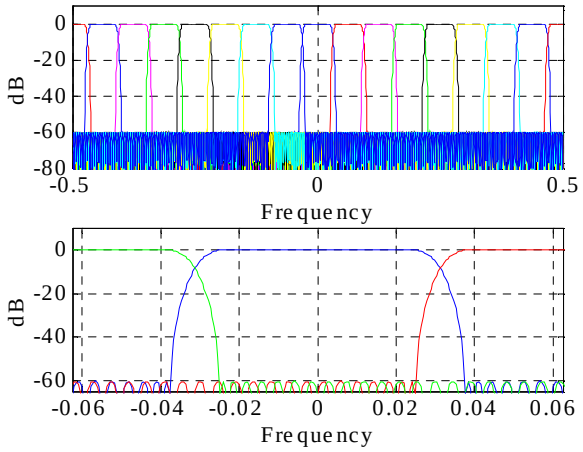
Recall that the IDFT of an M-point sequence $Y(k)$ is

$$y(n) = \sum_{k=0}^{M-1} Y(k) e^{j2\pi nk/M} \qquad n = 0,1,\ldots,M-1$$

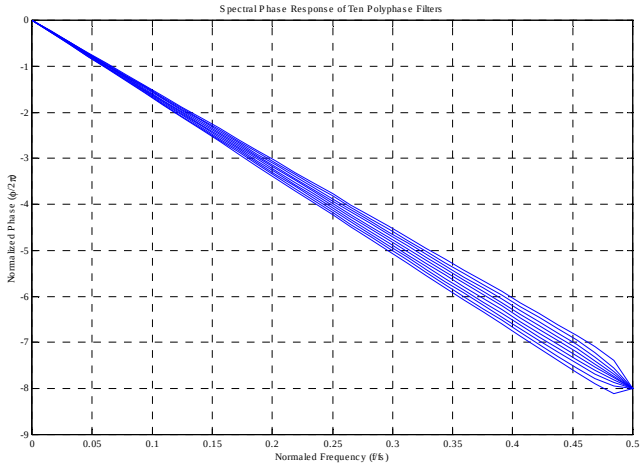
If the M phase rotators are sequenced over all of the M values of k we recognize that this is the same as computing an IDFT



Channelizer Filter Bank

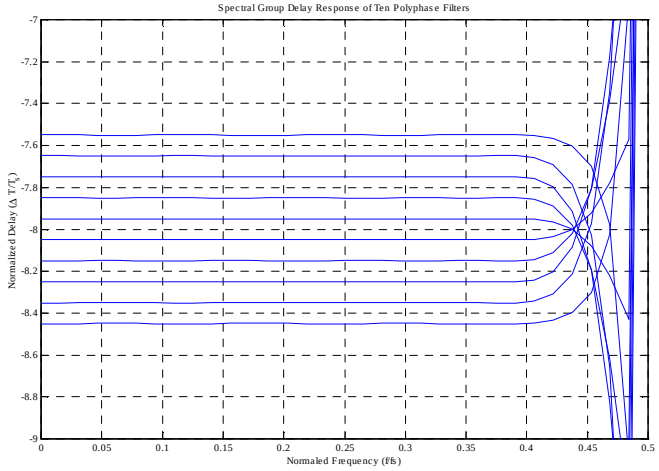


Phase Response of Paths in Ten Stage Polyphase Filter



Channelization 11

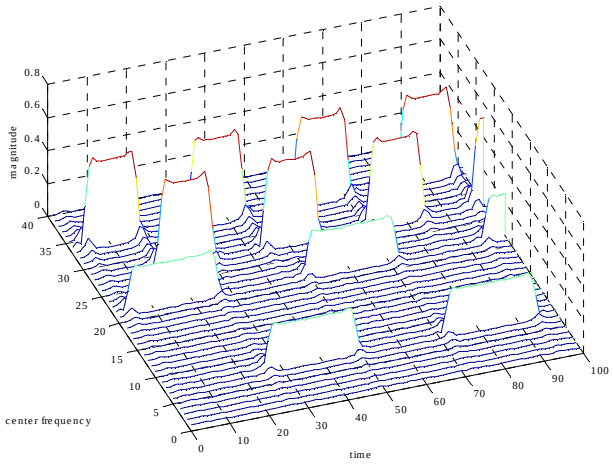
Group Delay of Paths in Ten-Stage Polyphase Filter



Channelization 12

40 Channel Polyphase Receiver

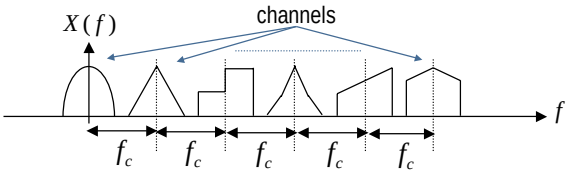
Time Series Waterfall from 40-Channel Polyphase Receiver



Channelization 13

Channelized Receiver Example

- Design a channelized receiver

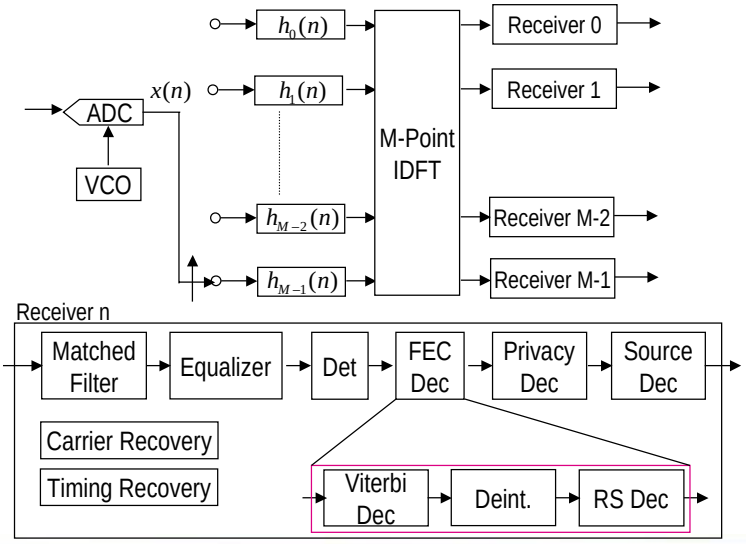


sample rate fs = 100 MHz, 12b samples
16 channels $f_c = 6.25$ MHz
M-ary QAM modulation
concatenated decoder

Filter requirements
60 dB stopband ripple
0.2 dB passband ripple

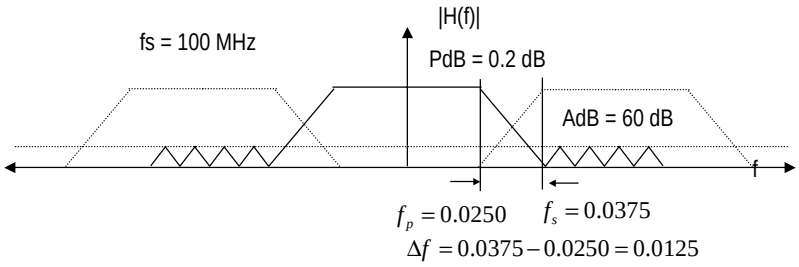
Channelization 14

Polyphase Channelizer



Channelizer Filter Design

- Design the prototype filter



$f_p = 0.0250 \quad f_s = 0.0375$

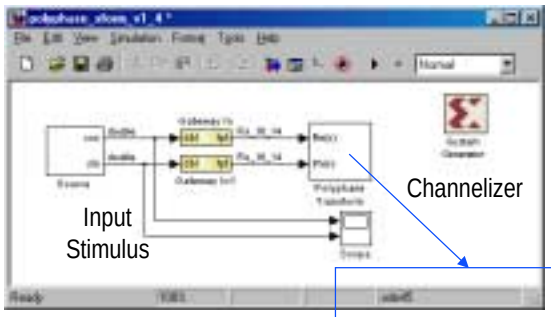
$\Delta f = 0.0375 - 0.0250 = 0.0125$

Filter length approximation due to Prof. fred harris
San Diego State University

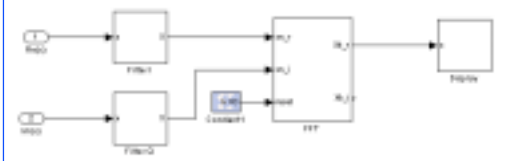
$$N \approx \frac{f_s}{\Delta f} \cdot \frac{AdB}{22}$$

$$= \frac{1}{0.0125} \cdot \frac{60}{22} = 219$$

Channelizer Implementation

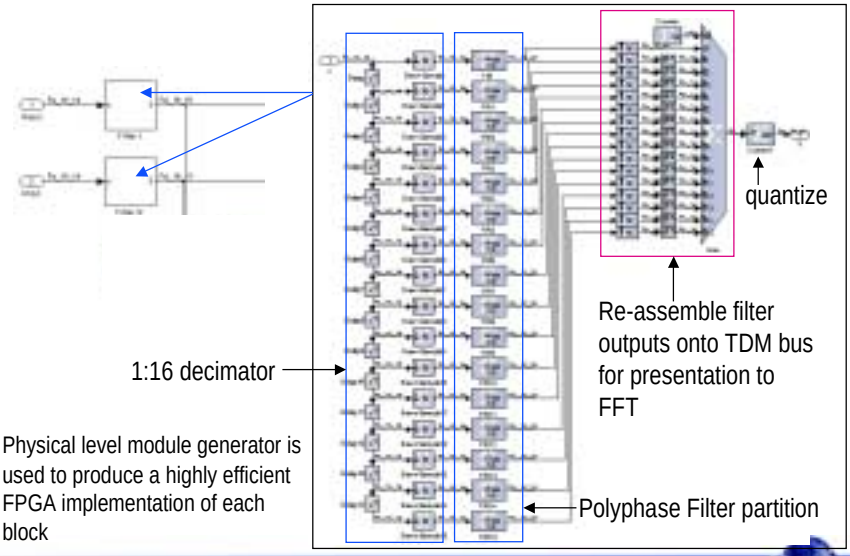


Capture data to Matlab workspace for post analysis/display



Channelization 17

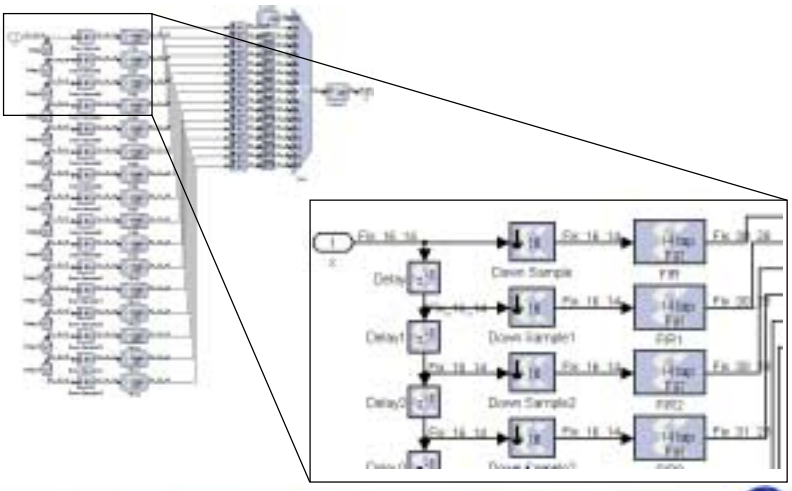
System Generator Implementation



Physical level module generator is used to produce a highly efficient FPGA implementation of each block

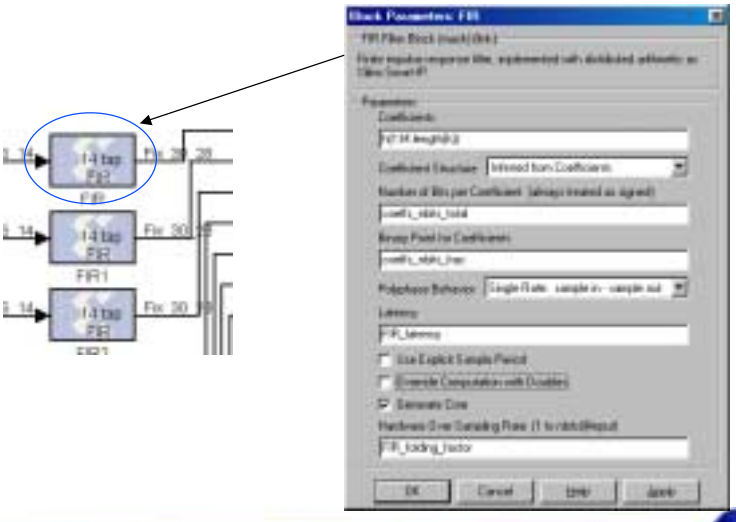
Channelization 18

System Generator Implementation



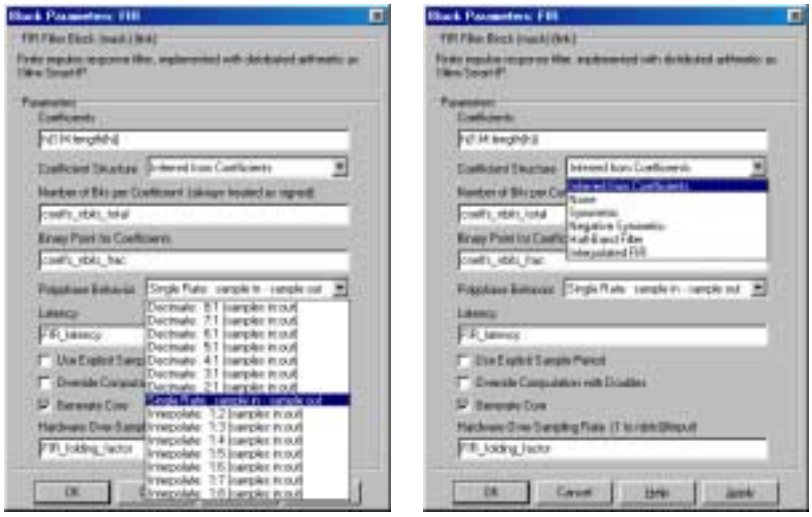
Channelization 19

System Generator Implementation



Channelization 20

Filter Module



Channelization 21

System Generator Implementation

-The *Hardware Over-Sampling Rate* (folding factor) field of the FIR filter block allows the designer to tradeoff throughput with FPGA area

-The filter throughput is fclk/R where R is the Hardware Over-Samlping Rate and fclk is the filter clock rate … which is not necessarily the same as the filter sample rate

-When R=1 a new filter output is generated on each clock cycle

-For R=2 a new output is generated every second clock cycle

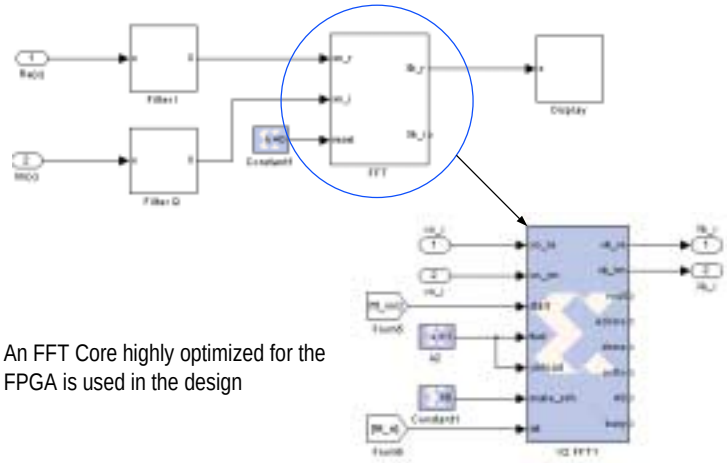
-With this control the designer can realize a filter that best matches the requirements of the system

Channelization 22

System Generator Implementation

- In this design the input sample rate is 100 MHz (12b samples)
- The sample rate presented to each filter in the filter bank is 100e6/16 = 6.25 MHz
- The most efficient implementation for the filters is to employ a folding factor equal to the input sample precision
- The filters will be clocked at a frequency of 12 x 6.25e6 = 75 MHz
- The digital clock manager (DCM) can be used to generate the various clock frequencies

System Generator Implementation



An FFT Core highly optimized for the FPGA is used in the design

Implementation Statistics

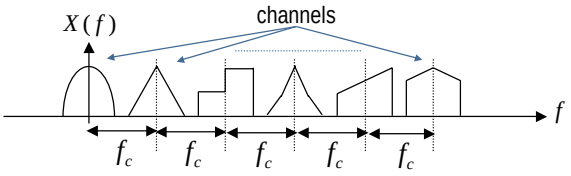
- Distributed arithmetic implementation employed in this example
- Virtex-II Pro 2vp50-7
  - Filter Bank: 6500 slices
  - fclk (max) = 200 MHz

$$\underbrace{2}_{\text{COMPLEX}} \times \underbrace{16}_{\text{RE-SAMPLING RATIO}} \times \underbrace{14}_{\text{SUB FILTER LENGTH}} \times \underbrace{200e^6}_{\text{CLOCK}} = 90 \text{ GMACs}$$

- 448 mpys @ 200 MHz would be required to meet this performance using a MAC FIR approach
- Use right algorithm for the problem
- Explore the FPGA/algorithmic design space

Channelized Receiver Example (2)

- Design a channelized receiver



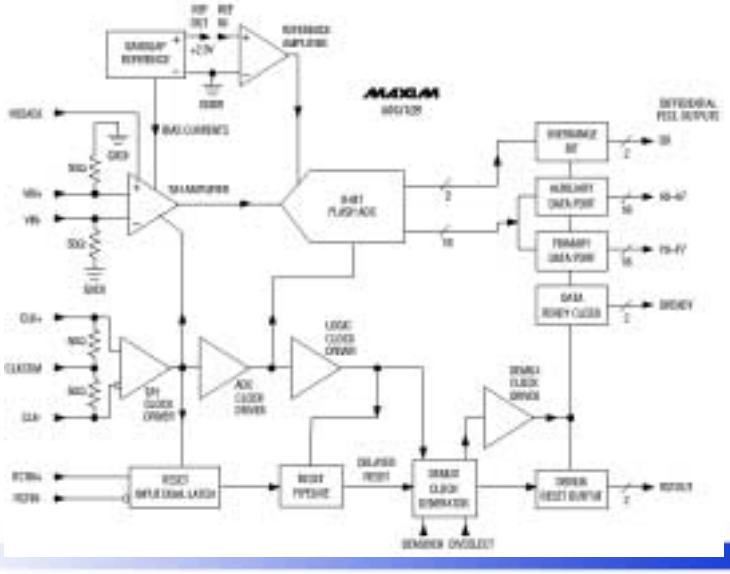
  - sample rate fs = 1 GHz, 12b samples
  - 16 channels
  - M-ary QAM modulation
  - concatenated decoder

Wide-Band Channelized Receiver

- In this case the challenge is to deliver the samples from the 1 Giga-sample/s ADC to the DSP engine
- Utilize the double data rate (DDR) capability of the Virtex-2 input/output blocks (IOBs)
- Use the DCM to generate the required clocks

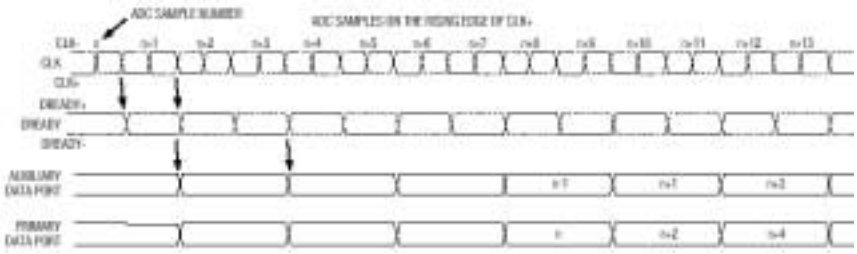
Channelization 27

High-Speed 1GHz ADC



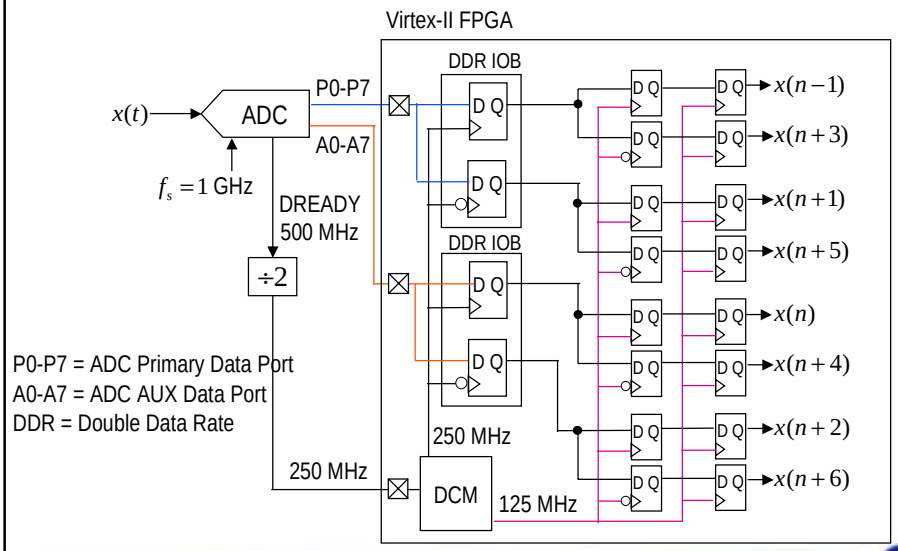
Channelization 28

ADC Timing



Channelization 29

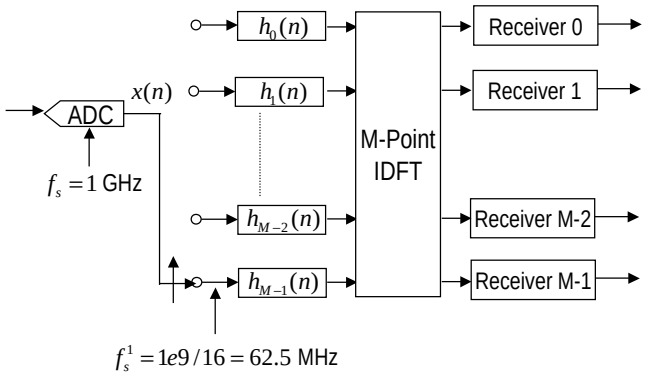
1GHz ADC-FPGA Interface



P0-P7 = ADC Primary Data Port
A0-A7 = ADC AUX Data Port
DDR = Double Data Rate

Channelization 30

1 GHz Channelizer



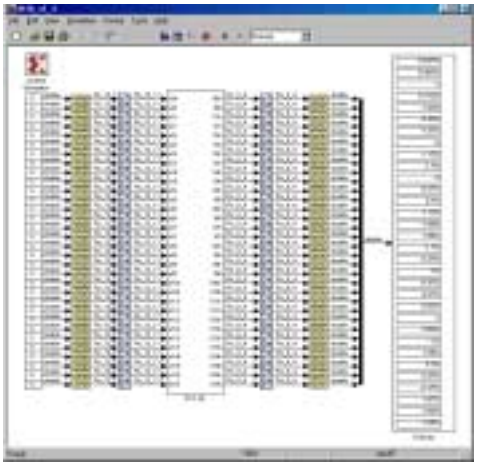
- Each polyphase sub-filter must support a throughput of 1e9/16 = 62.5 Ms/s
- A 1 Giga-sample/s 16-point FFT is required

Implementation Statistics

- Filter bank arithmetic requirements
  - 32 x 14 x 62.5e6 = 28 GMACs
- Distributed arithmetic filters used for polyphase filter bank
  - Hardware folding factor = 2
    - Each subfilter is allocated a 2 clock cycle schedule to execute
    - Filter bank fclk = 2 x 1e9/16 = 125 MHz
    - 2vP50-6
      - 11,155 slices 47% of the device
    - Interesting figure of merit: 28e9/11155 = 2.5 MMACs/logic slice

1 Giga-sample FFT in System Generator

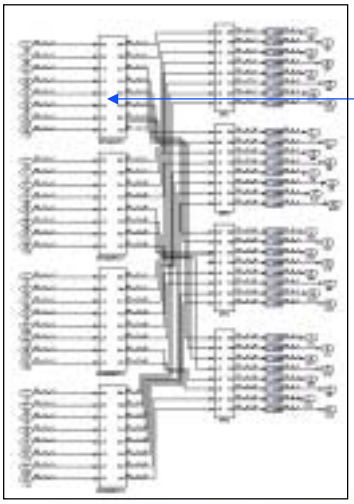
A very high-speed FFT is required for the channelizer
1 Giga-samples/s
Built in System Generator



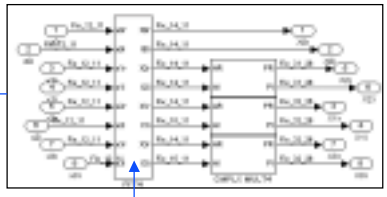
Channelization 33

1 Gs/s FFT in System Generator

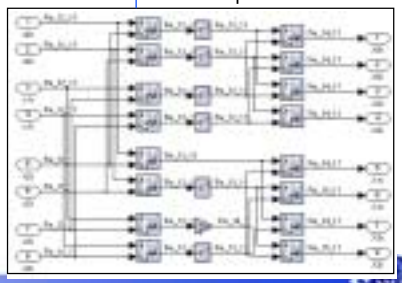
Butterfly network



Radix-4 butterfly



4-point FFT kernel



Channelization 34

Implementation Statistics

- Vectorized interface
  - 1 vector delivered/produced each clock cycle
- FPGA utilization
  - 1,812 slices
  - 36 embedded multipliers
- In this design required FFT fclk = 62.5 MHz
- One transform every 16 ns
- FFT will support fclk = 210 MHz
- 8 radix-4 dragonflies
  - 64 complex additions
    - 128 x 210e6 = 26.9 GOPs/sec.
  - 9 complex multiplications
    - 7.56 MMACs/sec.

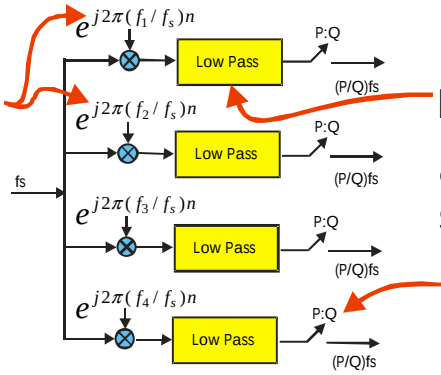
Modified Channelizer

- Conventional polyphase transform channelizer produces N maximally decimated output time series
- Many comm systems like to operate on multiple samples/symbol
  - e.g. many timing recovery loops in QAM demodulators
- Could interpolate each channelizer output time-series
- Alternative modify channelizer to embed (programmable) rate change

On Multichannel Receivers

With Arbitrary Uncoupled Selection of ...

Channel Spacing

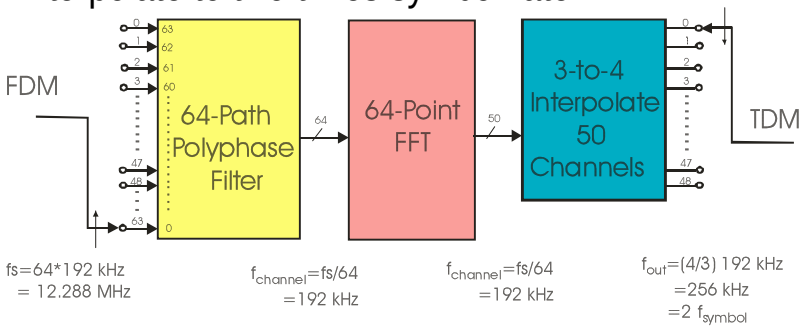


channel bandwidth

and channel sample rate

Conventional Channelizer Application

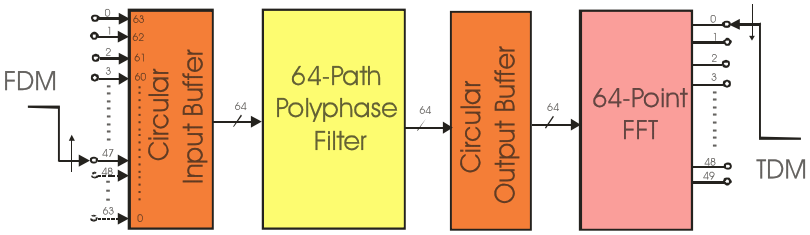
- Channelize
- Downsample to Nyquist rate
- Interpolate to two times symbol rate



Ratio of Input Rate to Output Rate = 12288/256 =48-to-1

Enhanced Channelizer Solution

- Replace Interpolator Function With Buffer Addressing



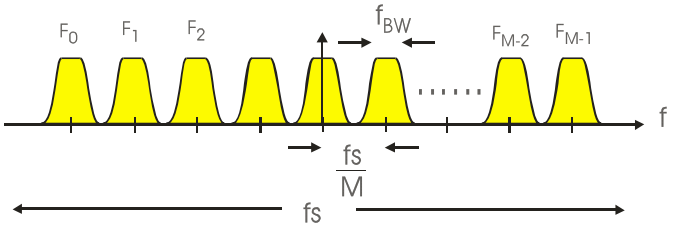
fs=64*192 kHz
= 12.288 MHz

$f_{channel}$=fs/64
=192 kHz

f_{out}=(4/3) 192 kHz
=256 kHz
=2 f_{symbol}

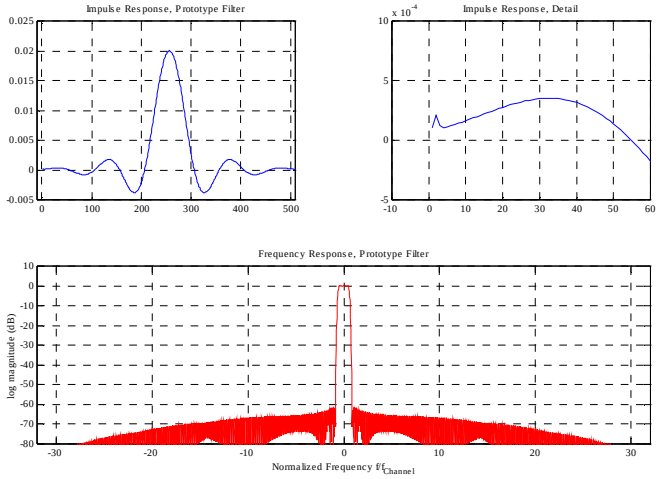
Ratio of Input Rate to Output Rate = 12288/256 =48-to-1

Performance Specifications for 50-Channel Polyphase Channelizer



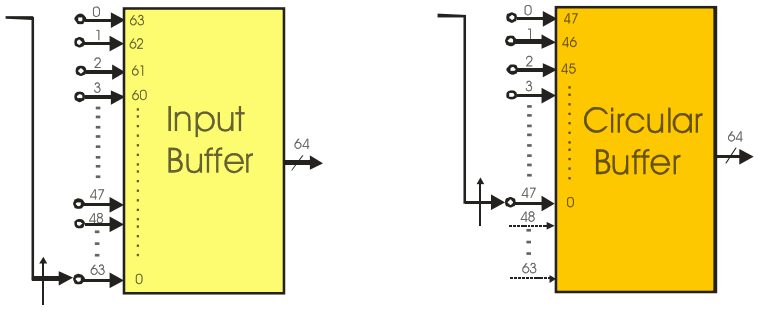
Symbol Rate	f_{sym}=128 KHz
Roll-Off	α=0.5
Two-Sided BW	f_{BW} =192 kHz
Channel Separation	f_{chan}=192 kHz
Number of Channels	M=64
Number of Useful Channels	N=50
Input Sample Rate 64*192	$f_{s\text{-}input}$ =12.288 MHz
Output Rate 2*128	$f_{s\text{-}output}$= 256 kHz

Time and Frequency response of Remez Filter Design with Modified End Points



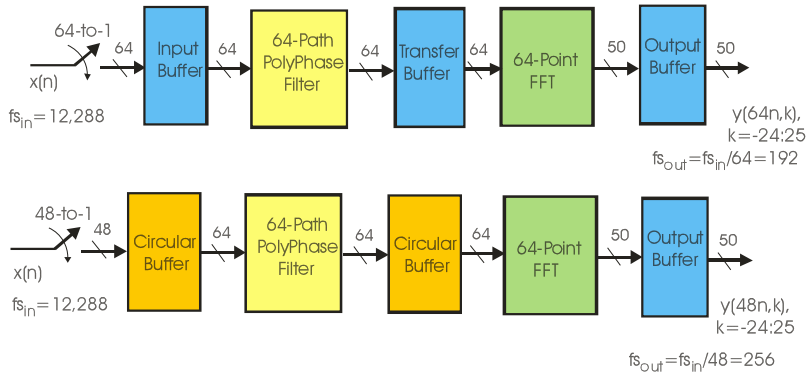
Channelization 41

Commutators for Standard Input Buffer and for Circular Input Buffer

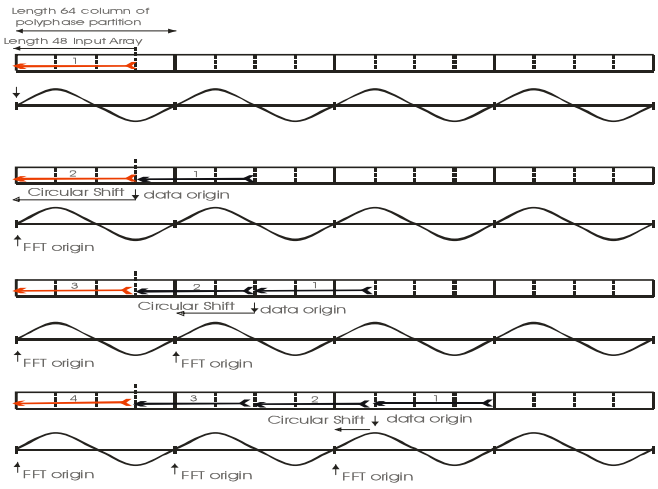


Channelization 42

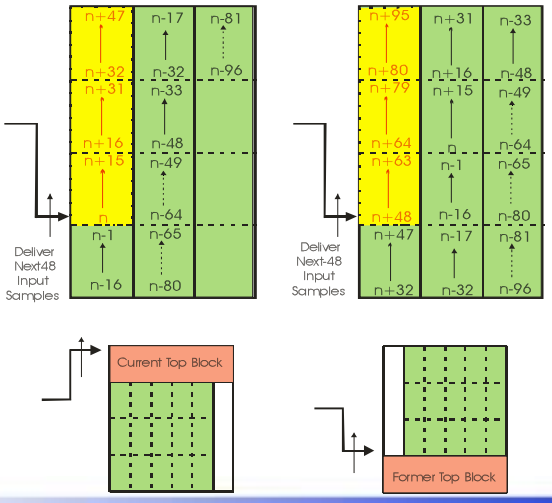
Standard Polyphase Channelizer and Modified Channelizer with Circular Buffers



Shifting Time Origin for Input Data of Polyphase Filter and of Resetting FFT

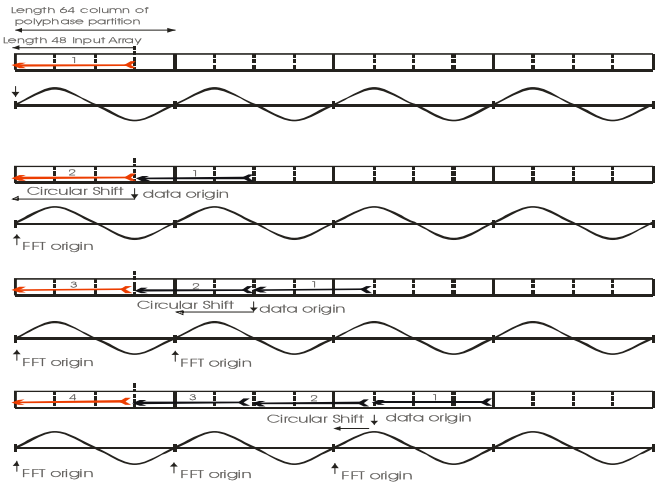


Content of 64-Point Circular Input Buffer for Two Successive 48 Point Input Blocks



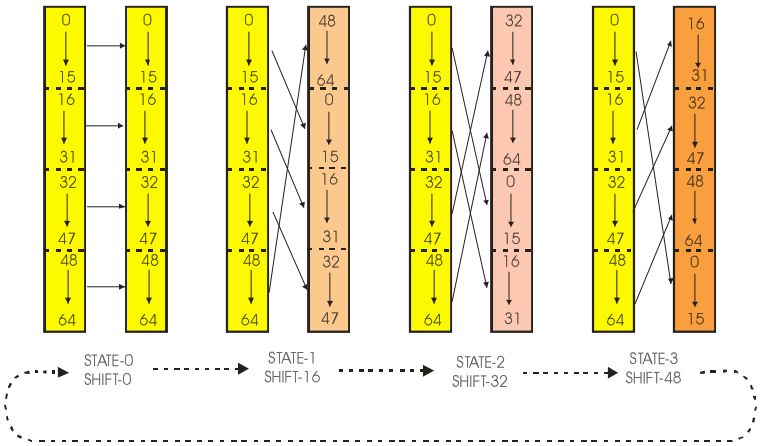
Channelization 45

Shifting Time Origin for Input Data of Polyphase Filter and of Resetting FFT



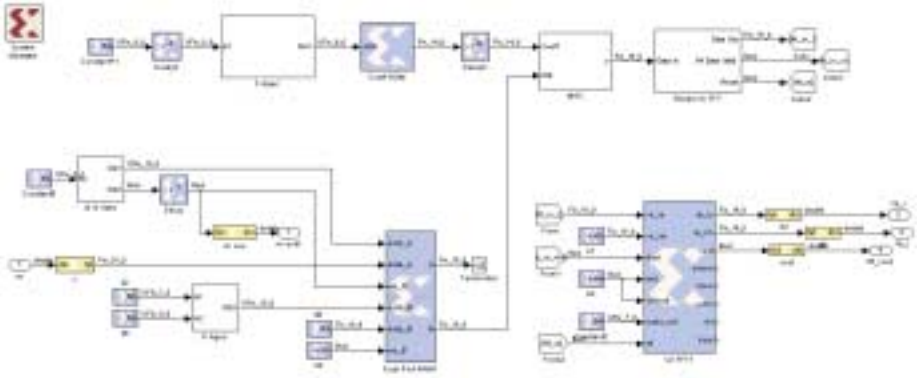
Channelization 46

Contents of Transfer Circular Buffer Aligning Origins for Successive Input Blocks



Channelization 47

Implementation



Channelization 48

Implementation Statistics

- 64 Channel channelizer
- Arbitrary re-sampling
  - 1000 slices
  - 5 embedded mpy
  - 6 block Virtex-II block memories