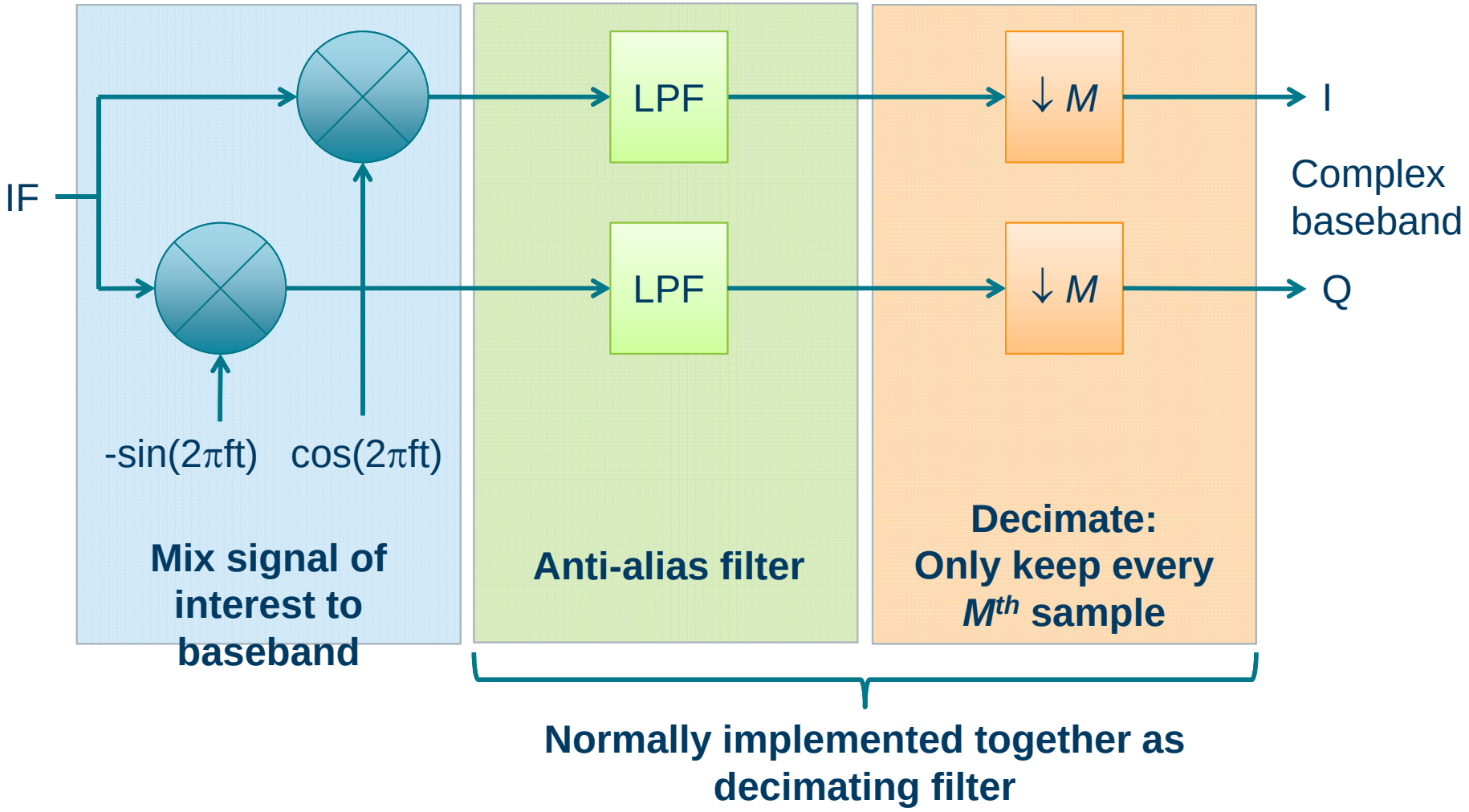
SIGNAL PROCESSING **IQ** : **FPGA** EXPERTISE

Efficient and Flexible Multi-channel Digital Up and Down Conversion for SDR and Cognitive Radio Applications

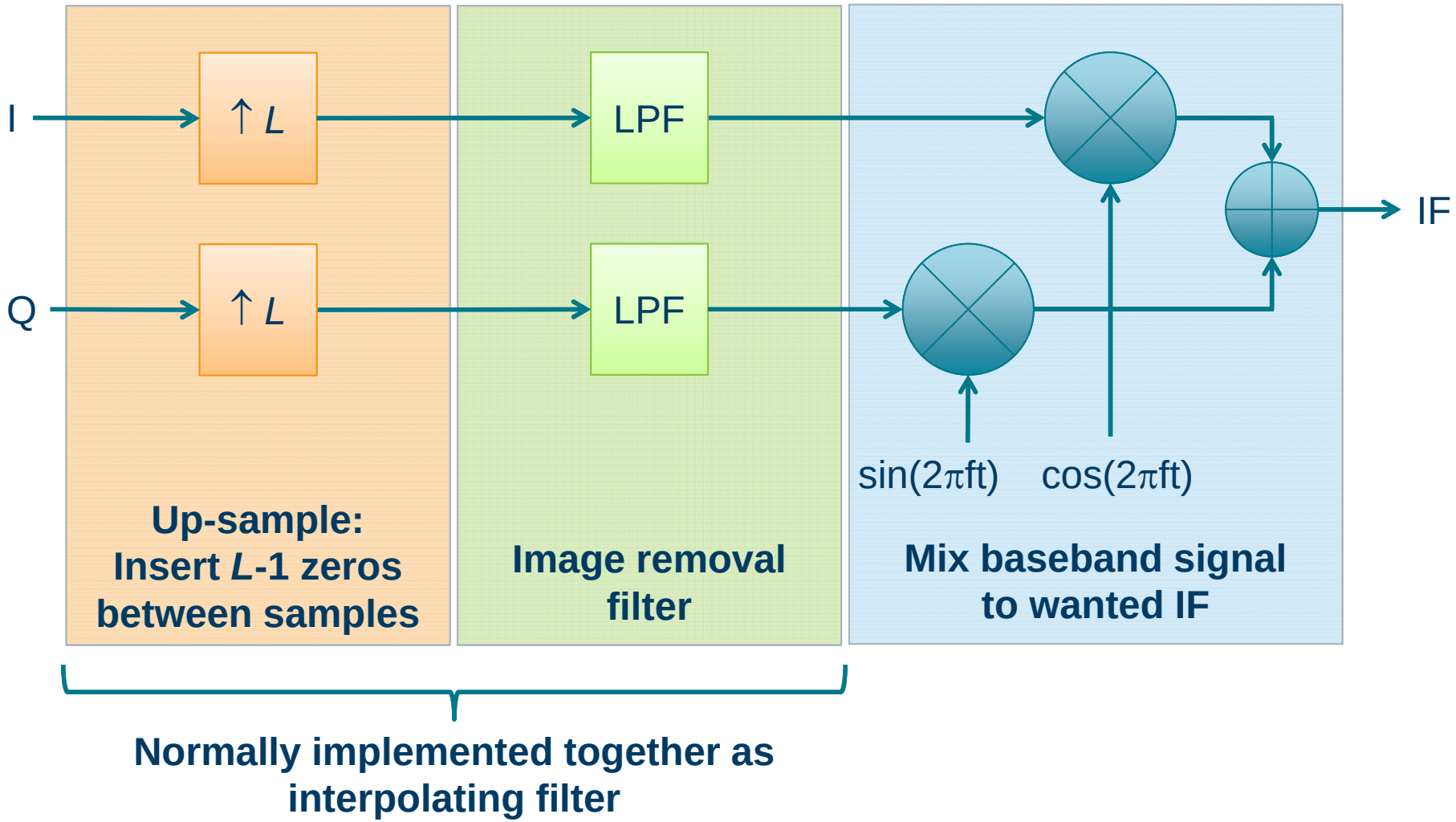
Eddy Fry, Technical Director, RFEL Ltd.

CONTENTS

DIGITAL DOWN CONVERTER (DDC)
CLASSIC ARCHITECTURE



DIGITAL UP CONVERTER (DUC)
CLASSIC ARCHITECTURE



FILTERS
CASCADED INTEGERATOR COMB

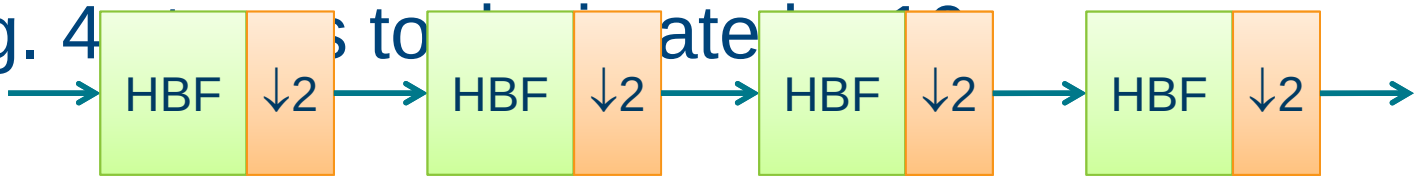
- CIC can be used for interpolation and decimation
- Traditionally used in DDC / DUC ASIC devices
- Advantages:
  - Efficient multi-rate architecture
    - » Constant processing load (independent of output rate for decimator and input rate for interpolator)
    - » Adder-based (no multiplications required)
      - Not an advantage for modern FPGAs implementations…
  - Any integer decimation / interpolation possible

- Disadvantages:
  - Large data-path bit-widths (depending on rate change)
    - » Truncation strategies can be used with care
  - Poor frequency response
    - » Narrow alias-free passband
    - » Large passband droop
    - » Normally cascaded FIR with additional decimation by 4 used to extract narrow alias-free band and compensate for droop
  - CIC + FIR provides rate changes of $4L$ *or* $4M$, where L is an integer interpolation and M is an integer decimation

DECIMATING / INTERPOLATING FILTERS

HALF BAND FILTER (HBF) #1

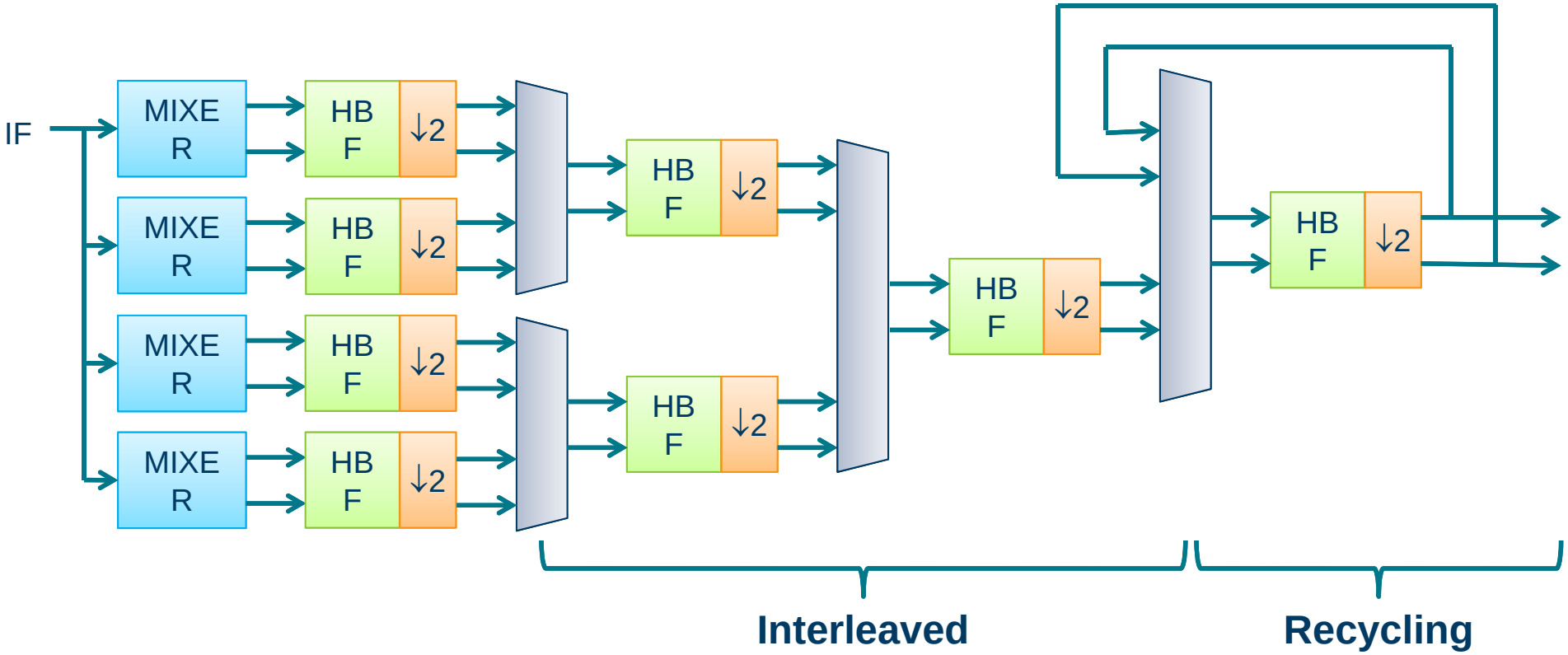
- HBF can be used to interpolate or decimate by two
- Coefficient zeros mean that filtering and rate change are combined in an efficient multi-rate architecture
- Cascaded to provide integer power of two rate changes
- E.g. 4 [illegible] to [illegible]ate by 16



  - Each stage processes at half the rate of the previous stage
  - Higher-rate stages require shorter filters due to low bandwidth to sample-rate ratio

DECIMATING / INTERPOLATING FILTERS
HALF BAND FILTER (HBF) #2

- Can be interleaved and re-cycled to provide increased efficiency in multi-channel applications



DECIMATING / INTERPOLATING FILTERS

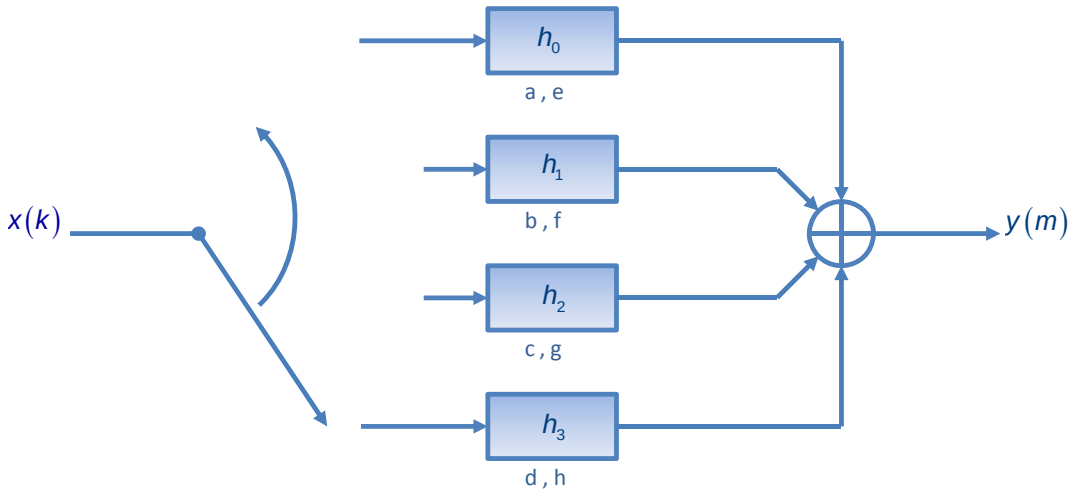
HALF BAND FILTER (HBF) #3

- Advantages:
  - Very efficient multi-rate architecture
  - Can be interleaved / re-cycled to increase overall efficiency
- Disadvantages:
  - Rate changes limited to integer powers of two

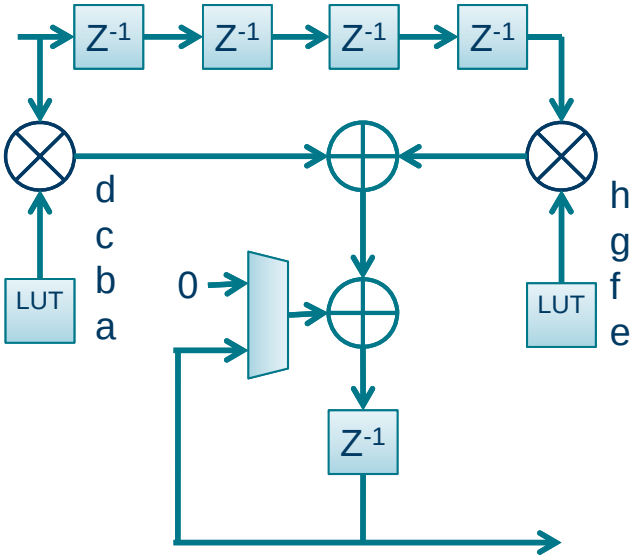
DECIMATING / INTERPOLATING FILTERS

MULTI-RATE FIR #1

- Multi-rate filters can be used for interpolation and decimation
- For decimator; only calculate the wanted output samples (not the discarded samples)
- 8-tap decimate by 4 example



Polyphase Form

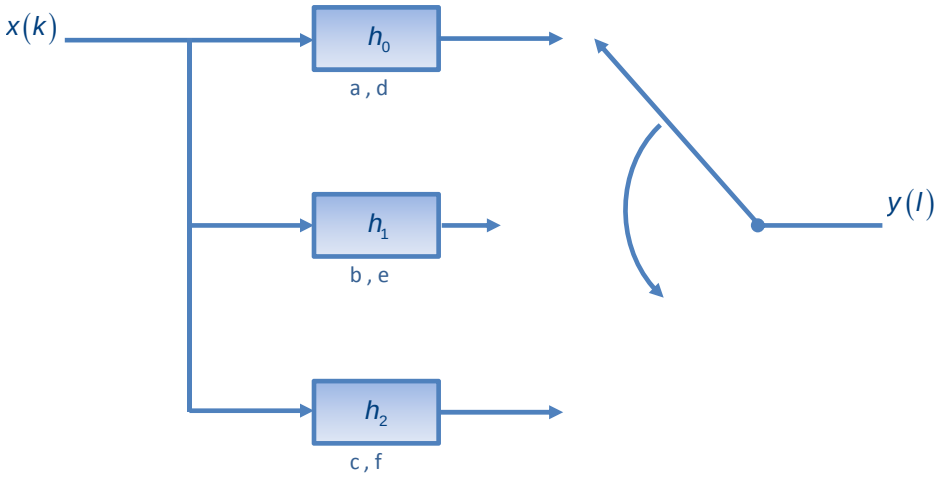


Interleaved Architecture

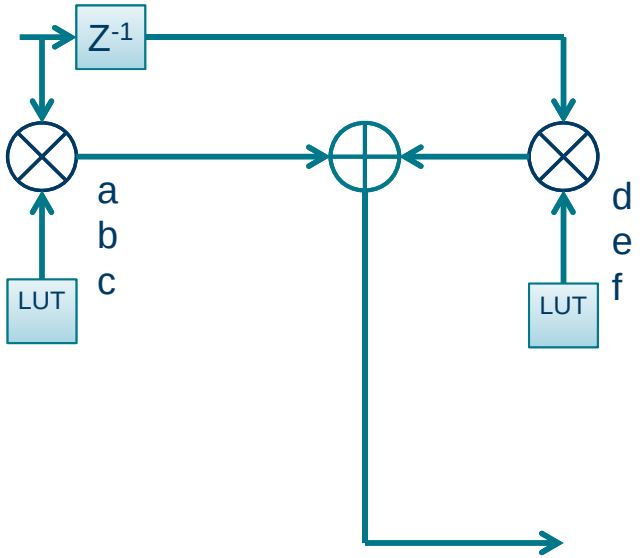
DECIMATING / INTERPOLATING FILTERS

MULTI-RATE FIR #2

- For Interpolator; filter calculations using only input samples (not up-sample zeros)
- 6-tap interpolate by 3 example



Polyphase form



Interleaved Architecture

DECIMATING / INTERPOLATING FILTERS

MULTI-RATE FIR #3

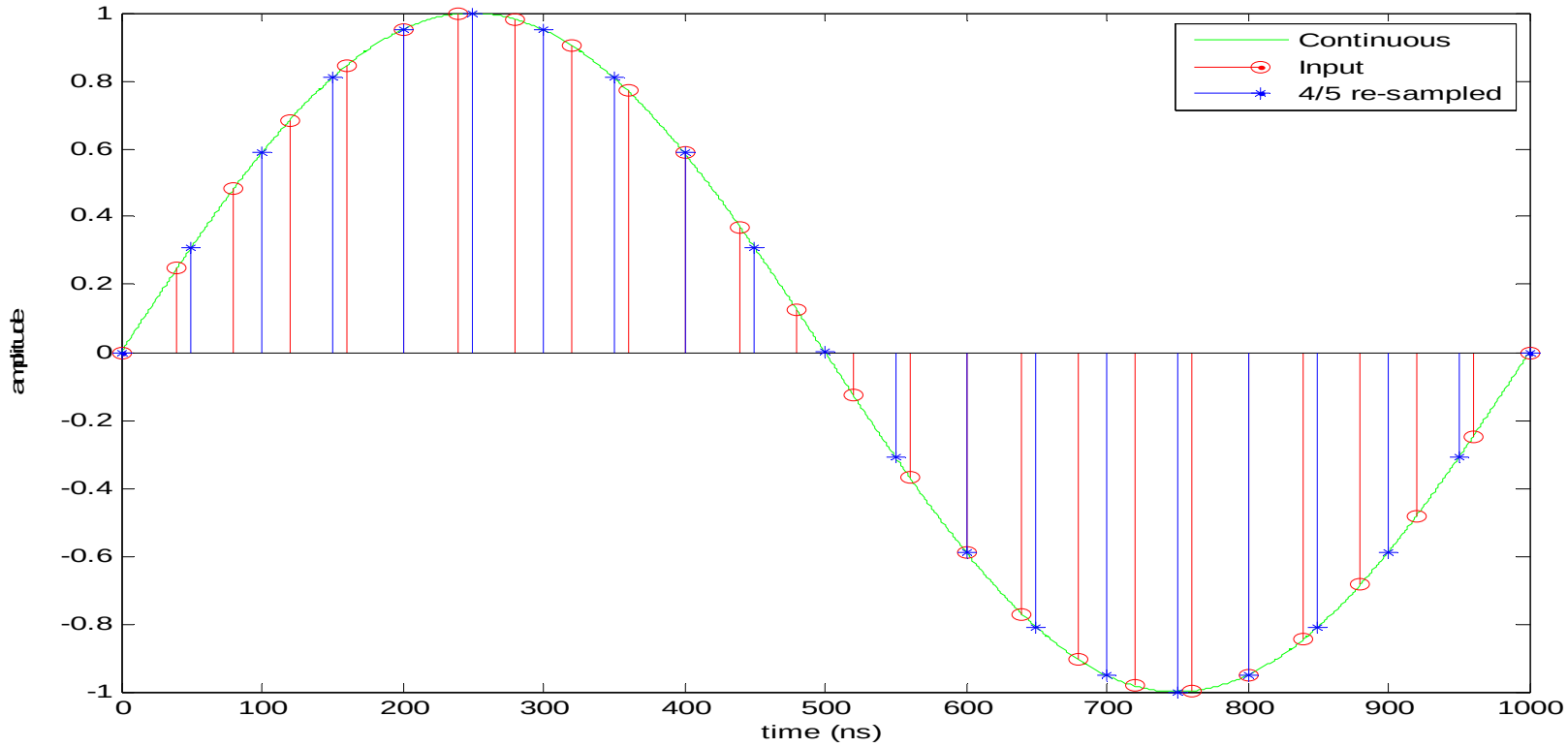
- Advantages:
  - Complete control of frequency response
  - Processing load depends on input rate for decimator or output rate for interpolator
  - Uses regular architecture structures (MAC, systolic array etc.)
  - Integer decimations or interpolations
  - Run-time programmable rate change possible
- Disadvantages:
  - Different coefficient set required for each rate change
  - Very long filters for large rate changes

DECIMATING / INTERPOLATING FILTERS
FRACTIONAL RATE CHANGE #1

- Can interpolation *and* decimation be efficiently combined to provide non-integer rate changes?

DECIMATING / INTERPOLATING FILTERS

FRACTIONAL RATE CHANGE #2



DECIMATING / INTERPOLATING FILTERS
FRACTIONAL RATE CHANGE #3

- Possible implementations
  - Combined polyphase structure
    - » Good for small integer ratios
  - Interpolation of over-sampled data
    - » $Sinc^2$ frequency response for linear interpolation
    - » Linear is easy to implement, but inherently inefficient due to degree of oversampling required for good performance
  - Farrow structure
    - » High-order interpolation algorithm
    - » Can be hard to control frequency response, especially when challenging specification (high stopband attenuation for example)
  - RFEL's proprietary multi-rate architecture…

DECIMATING / INTERPOLATING FILTERS
FRACTIONAL RATE CHANGE #4

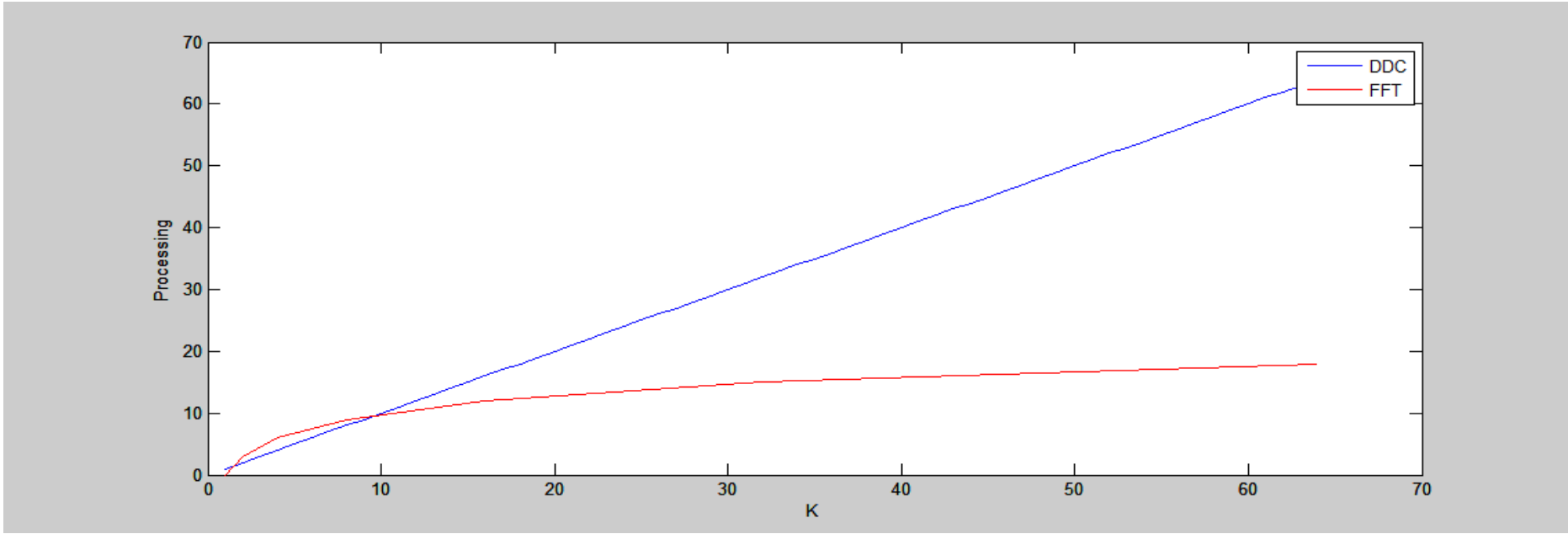
- RFEL's proprietary multi-rate architecture provides:
  - Equivalence to Matlab's 'upfirdn' function
  - Can support very large ***runtime-programmable*** integers ($>2^{32}$)
  - Virtually any rate change (sub-Hz resolution)
  - Arbitrary frequency response (>100dB SFDR is 'easy')
  - Both increase and decrease of sample rate possible with the same fixed architecture
  - Independent processing of arbitrarily interleaved channels
  - Optimised for ASIC or FPGA

FFT-BASED ‘CHANNELISERS’

- ‘Channeliser’ defined here as a number of DDC channels
- Fast Fourier Transform (FFT) is an efficient (fast) implementation of Discrete Fourier Transform (DFT)
- A DFT can be used to split a band into equal sections
  - Called a ‘uniformly-distributed analysis filter bank’
- IFFT can be used to combine equal sections
  - Called a ‘uniformly-distributed synthesis filter bank’
- Functionally equivalent to a bank of DDCs or DUCs with:
  - Fixed raster of mix frequencies
  - Identical decimation
  - Identical baseband frequency response

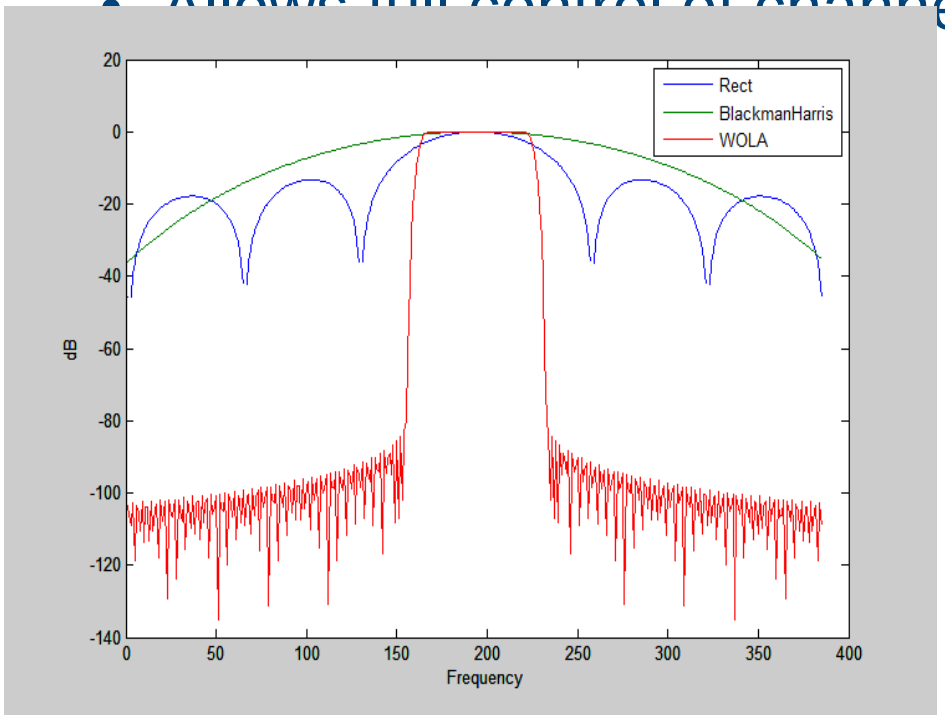
FFT-BASED CHANNELISERS
MOTIVATION

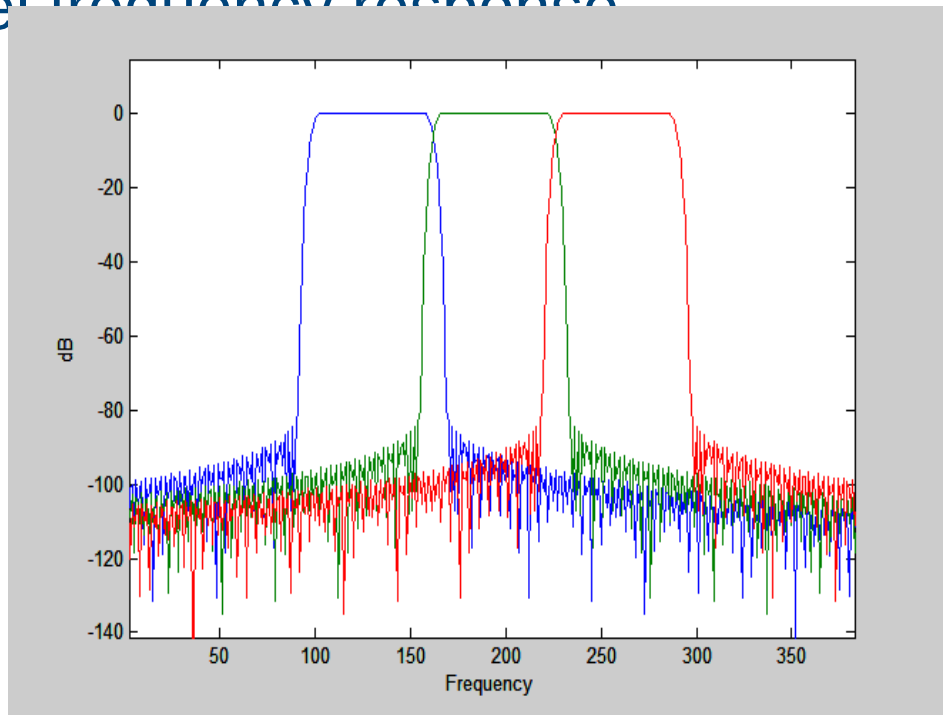
- Why is this interesting?
- A bank of discrete DDCs require O (K^2) processing operations
  - K is number of DDC channels
- Most FFT algorithms only require O (K log K) processing operations



FFT-BASED CHANNELISERS
WINDOWING

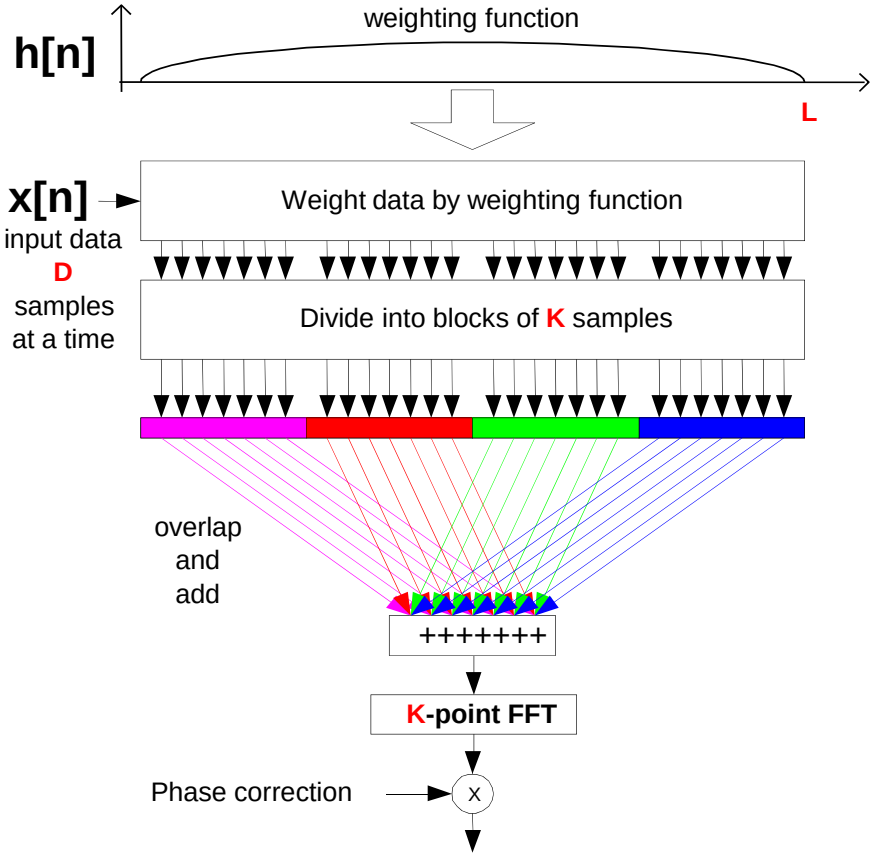
- Weight-OverLap and Add (WOLA) can be combined with an FFT to achieve an arbitrary frequency response
  - Allows full control of channel frequency response

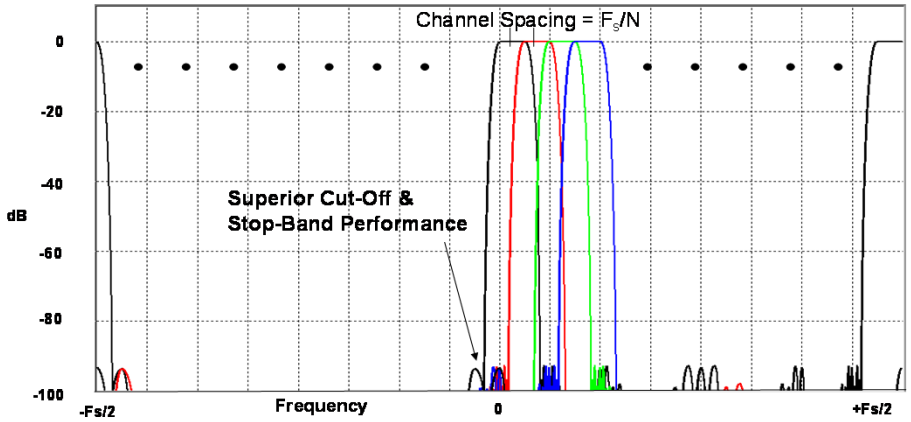




FFT-BASED CHANNELISERS
WOLA-FFT ARCHITECTURE

- WOLA implemented as interleaved polyphase structure
- DFT implemented as pipeline FFT



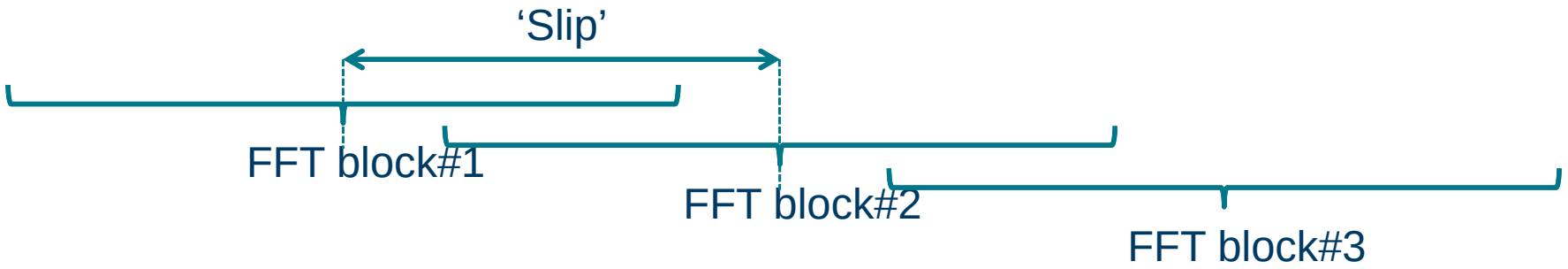


FFT-BASED CHANNELISERS
FFT CHANNEL FREQUENCIES

- Channel frequencies equally spaced at f_s / K
  - f_s is input sample rate
  - K is FFT length
- Achieve required spacing by selecting suitable f_s and K
  - K usually integer power of two (2,4,8 etc.)
    - » Can build ‘mixed-radix’ FFTs using other integer factors (3, 5, 7 etc.)
  - f_s often constrained by system (bandwidth, IF, ADC etc.)
    - » Can re-sample using rational rate re-sampler
- GMR-1 (Thuraya satellite) example:
  - f_s = 192MS/s, K = 6144 (6*1024), spacing = 31.25kHz

FFT-BASED CHANNELISERS
FFT CHANNEL SAMPLE RATE

- WOLA can be used to achieve integer decimations by adjusting the Overlap
  - ‘Slip’ data by an integer number of samples for each successive FFT ‘block’



- Convenient to slip by K/2 samples (50% overlap)
  - Over-sampling ensures alias-free passband
  - Simple implementation
  - Phase ‘de-spin’ simple

FFT-BASED CHANNELISERS
FFT CHANNEL FRACTIONAL SAMPLE RATE

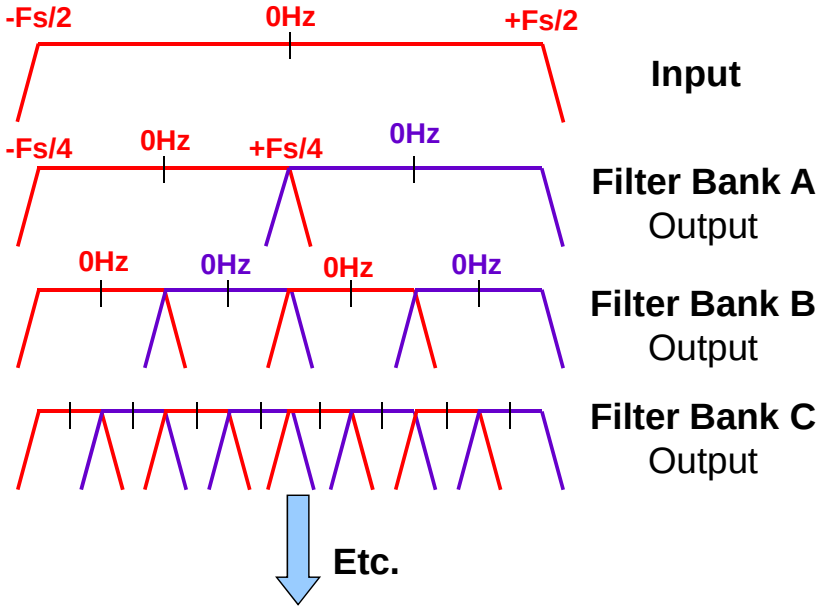
- Virtually any sample rate can be achieved by:
  - Use fractional rate re-sampler on output channels
  - Use integer slips other than K/2
  - RFEL has WOLA that can slip by fractional integer ratios…
- GMR-1 example:

  f_s = 192MS/s, K = 6144, sample rate = 93.56kS/s (4x23.4kS/s)

  1. WOLA slip = 3072 (K/2) + re-sampler with L = 936, M = 625
  2. WOLA slip = 81920/39 (fractional)

MULTI-RESOLUTION CHANNELISERS
PFT

- RFEL's Pipelined Frequency Transform (PFT) provides hierarchical channelisation by successive up and down mixes, filtering and decimation



- All levels in the hierarchy are available at the same time

MULTI-RESOLUTION CHANNELISERS
OTHER METHODS

- Other multi-resolution filter banks
  - Parallel WOLA-FFTs
    - » OK for a few resolutions, but inefficient for more
  - Quadrature mirror filter bank
    - » Limited frequency response – can’t overlap channels
  - Frequency-domain approaches
    - » Powerful but complex to implement

CHANNELISER MERITS

- Properties that define ‘flexibility’
  - Tuning
    - Frequency range
    - Frequency accuracy / resolution
    - Run-time programmable?
  - Output sample rate
    - Range
    - Accuracy / resolution
    - Run-time programmable?
  - Filtering
    - Channel independence
    - Run-time programmable?

DISCRETE DDC / DUC MERITS

- Tuning
  - » Frequency range — Excellent
  - » Frequency accuracy / resolution — Excellent
  - » Run-time programmable? — Excellent
- Output sample rate
  - » Range — Normally excellent
  - » Accuracy / resolution — Good (depending on architecture)
  - » Run-time programmable? — Excellent
- Filtering
  - » Channel independence — Excellent
  - » Run-time programmable — Excellent

WOLA-FFT CHANNELISER MERITS

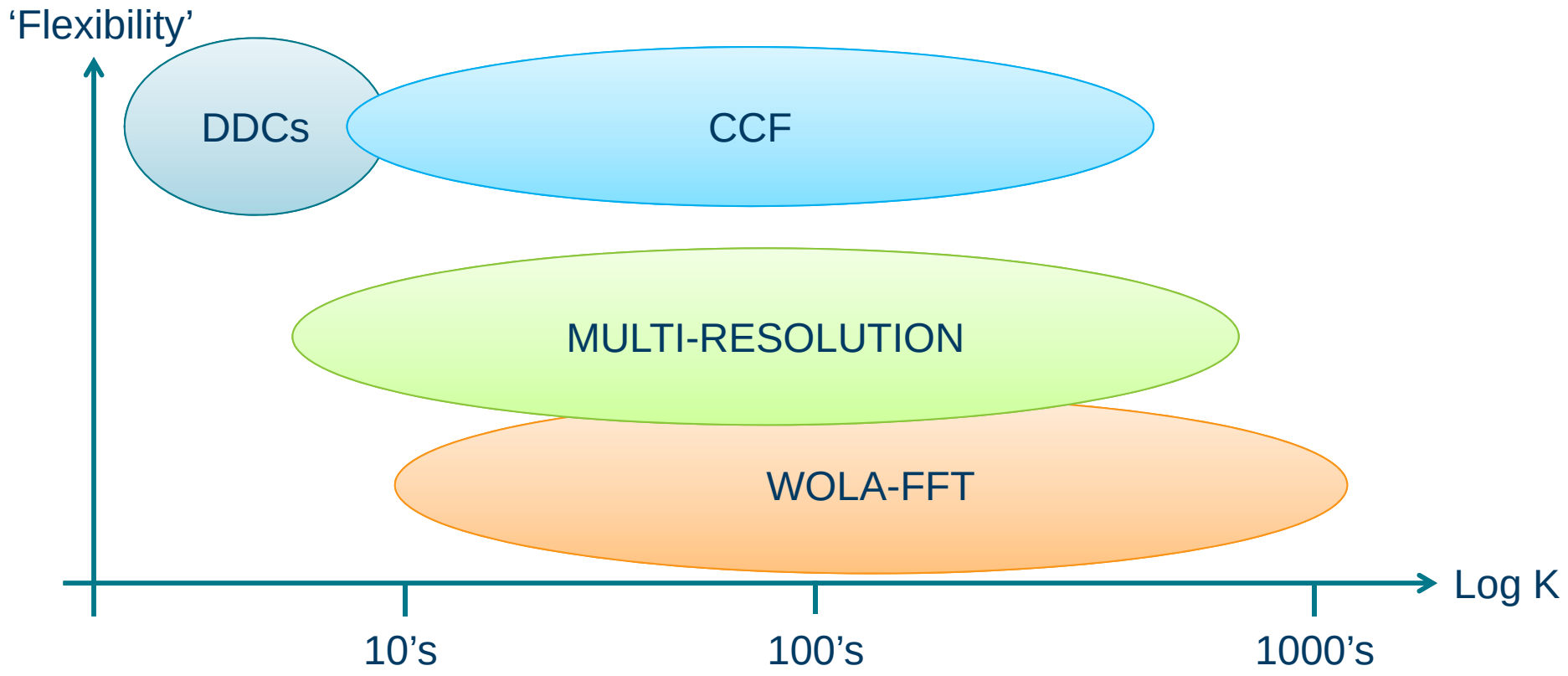
- Tuning
  - » Frequency range — Excellent
  - » Frequency accuracy / resolution — Poor (fixed-raster)
  - » Run-time programmable? — Poor (fixed)
- Output sample rate
  - » Range — Poor (maximum rate α 1/K)
  - » Accuracy / resolution — Poor (all the same!)
  - » Run-time programmable? — Poor to Good (depends on design)
- Filtering
  - » Channel independence — Poor (all the same)
  - » Run-time programmable — Poor (fixed)

MULTI-RESOLUTION CHANNELISER MERITS

- Tuning
  - » Frequency range — Excellent
  - » Frequency accuracy / resolution — OK (fixed-raster, but multi-res)
  - » Run-time programmable? — OK (fixed, but can select res)
- Output sample rate
  - » Range — OK (select nearest resolution)
  - » Accuracy / resolution — Poor (coarse selection)
  - » Run-time programmable? — OK
- Filtering
  - » Channel independence — OK (all the same at each res)
  - » Run-time programmable — OK (fixed, but can select res)

CHANNELISER MERITS

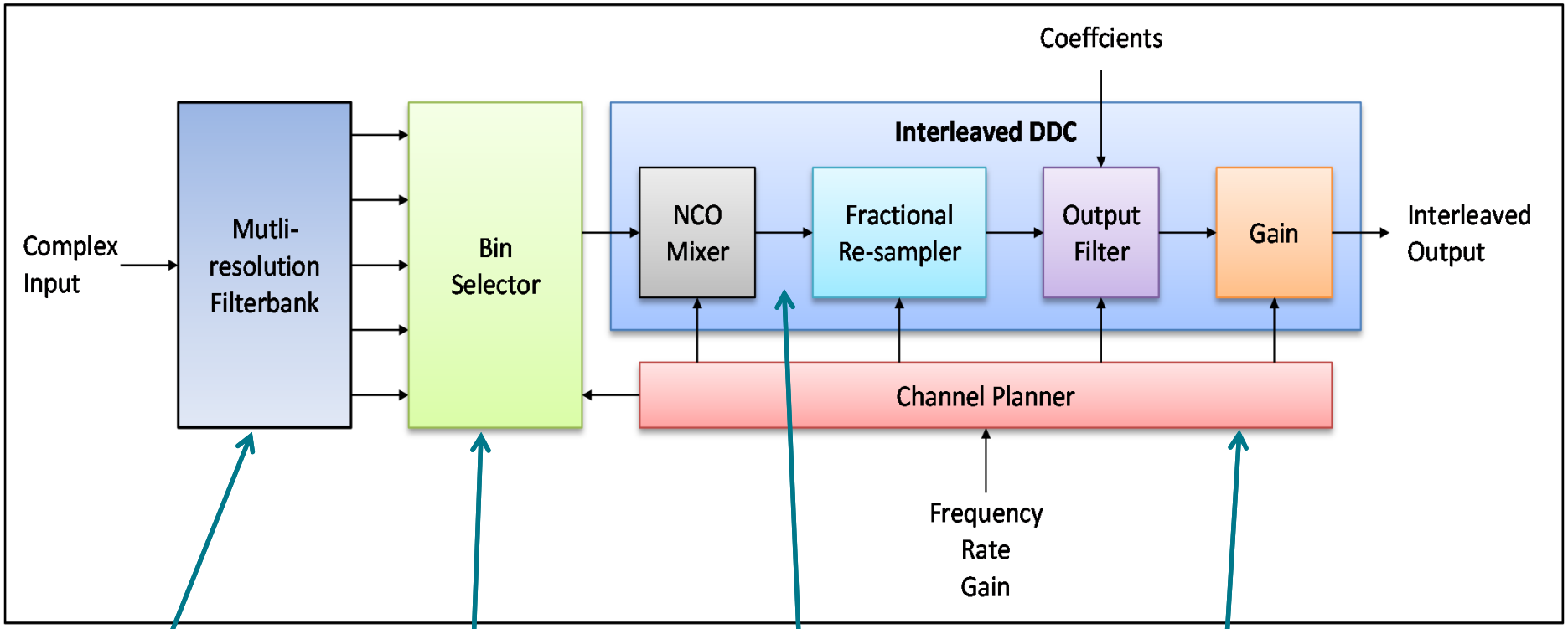
- Plot flexibility against a practical number of channels in a low-cost FPGA



CHANNEL CORE FLEX
DDC FLEXIBILITY *AND* O (K log K) COMPLEXITY

- ChannelCore Flex is RFEL's flagship flexible channeliser
- Designed primarily for FPGA (Xilinx and Altera)
- Combination of techniques previously discussed achieves high levels of flexibility, channel density and efficiency
- Behaves like a bank of fully independent DDCs
- Each channel can be run-time programmed for
  - Frequency anywhere in the input band to sub-Hz accuracy
  - Output sample rate from $f_s/2$ to few Hz with sub-Hz accuracy
  - Choice of 8 (or more if required) programmable output filters
  - Gain
- Only constraint is aggregate output sample rate must be < core clock rate
- Higher parallelism is possible for higher aggregate rates

CHANNEL CORE FLEX ARCHITECTURE



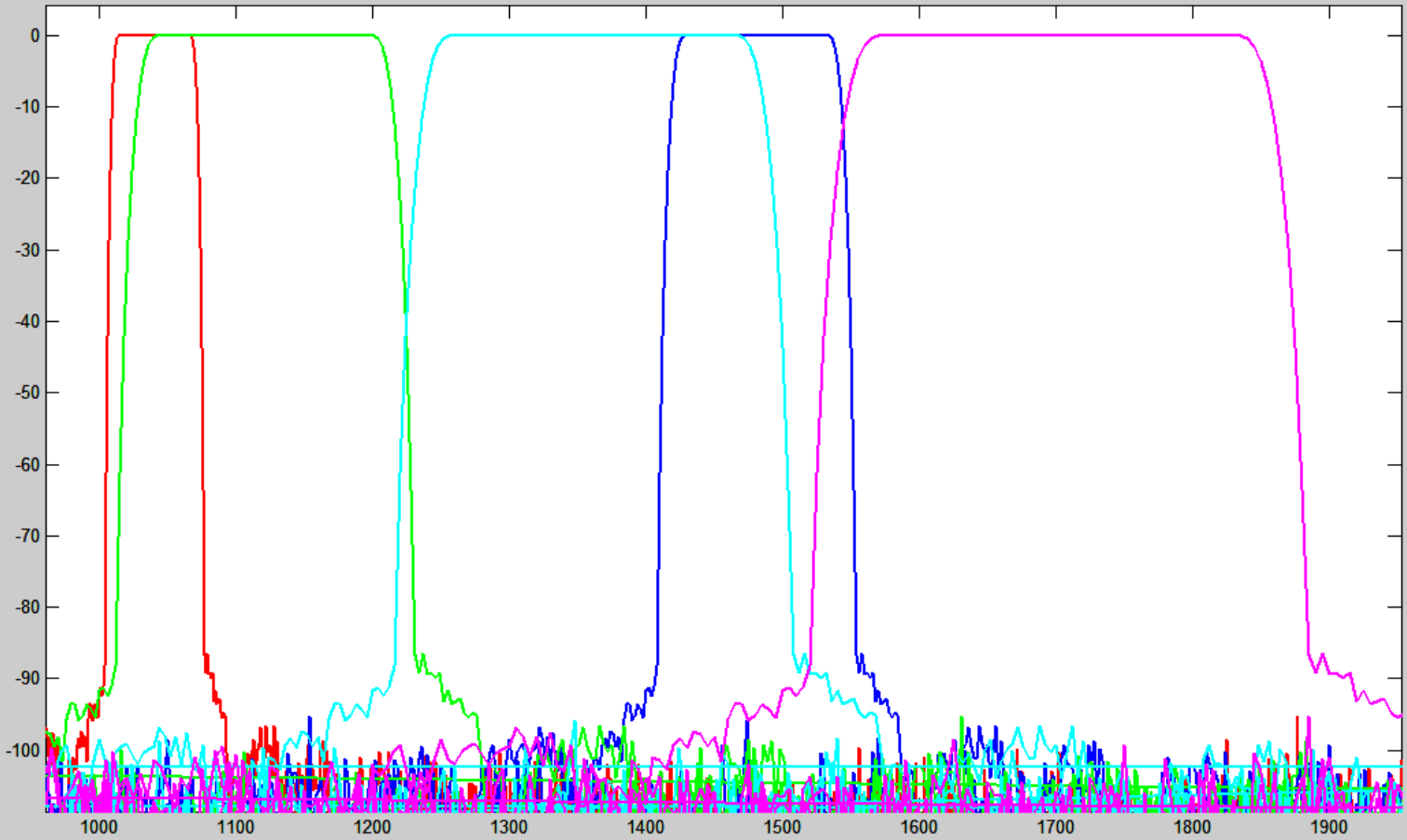
Coarse channelisation e.g. WOLA-FFT, PFT etc.

Select and Interleave bins closest to required freqs. and rates

Interleaved DDC applies fine frequency, rate, gain and filtering

Convert from Hz and dB to internal controls (NCO increments etc.)

CCF
EXAMPLE RESPONSE



CHANNEL CORE FLEX
MULTI-CHANNEL DUC

- The ChannelCore Flex architecture can be ‘inverted’ to provide efficient multi-channel up-conversion
- Most low-level modules for up-conversion already exist
  - Shaping filter is the same
  - Gain module is the same
  - Re-sampler increases rate (DDC decreases)
  - NCO mixer is the same
  - Bin selection is similar
  - Multi-resolution filter bank must synthesise

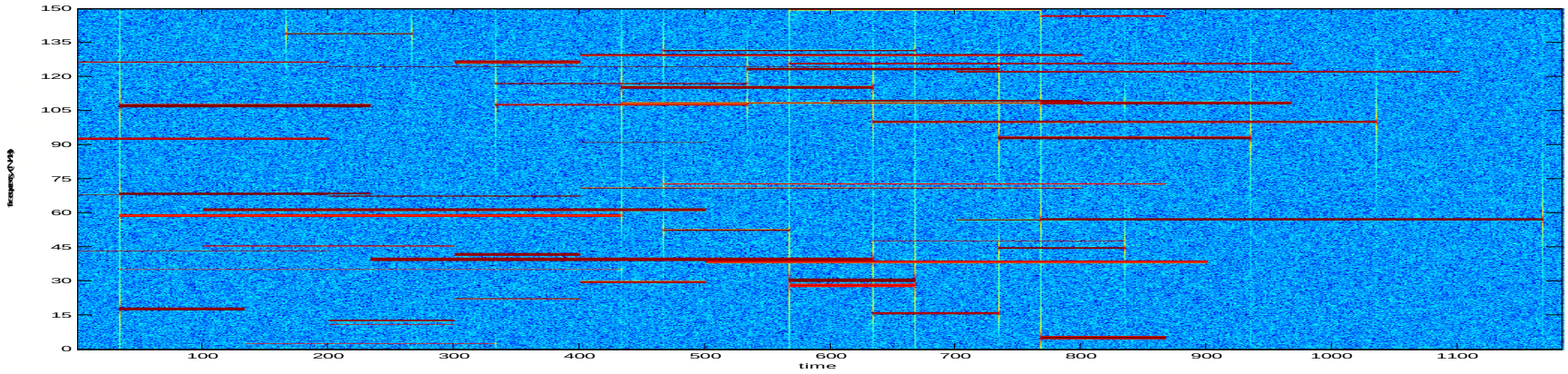
REAL-WORLD EXAMPLES
‘STANDARD’ CCF

- Specification:

• Number of complex inputs	2 (16-bit I&Q)
• Input sample rate	128MS/s
• Number of independent DDC channels	256
• Maximum output sample rate	64MS/s
• Minimum output sample rate	10kS/s
• Maximum aggregate output sample rate	128MS/s
• Tuning resolution	< 0.1Hz
• Output sample rate resolution	< 0.1Hz
• Alias-free passband width	0.8 x output rate
• SFDR	> 80dB
• Passband ripple	< 0.2dB pk-pk
• Number of output filter taps	32
• Number of output filter sets	8

- Core fits within a Xilinx Spartan-6 LX100 (<£100 part)

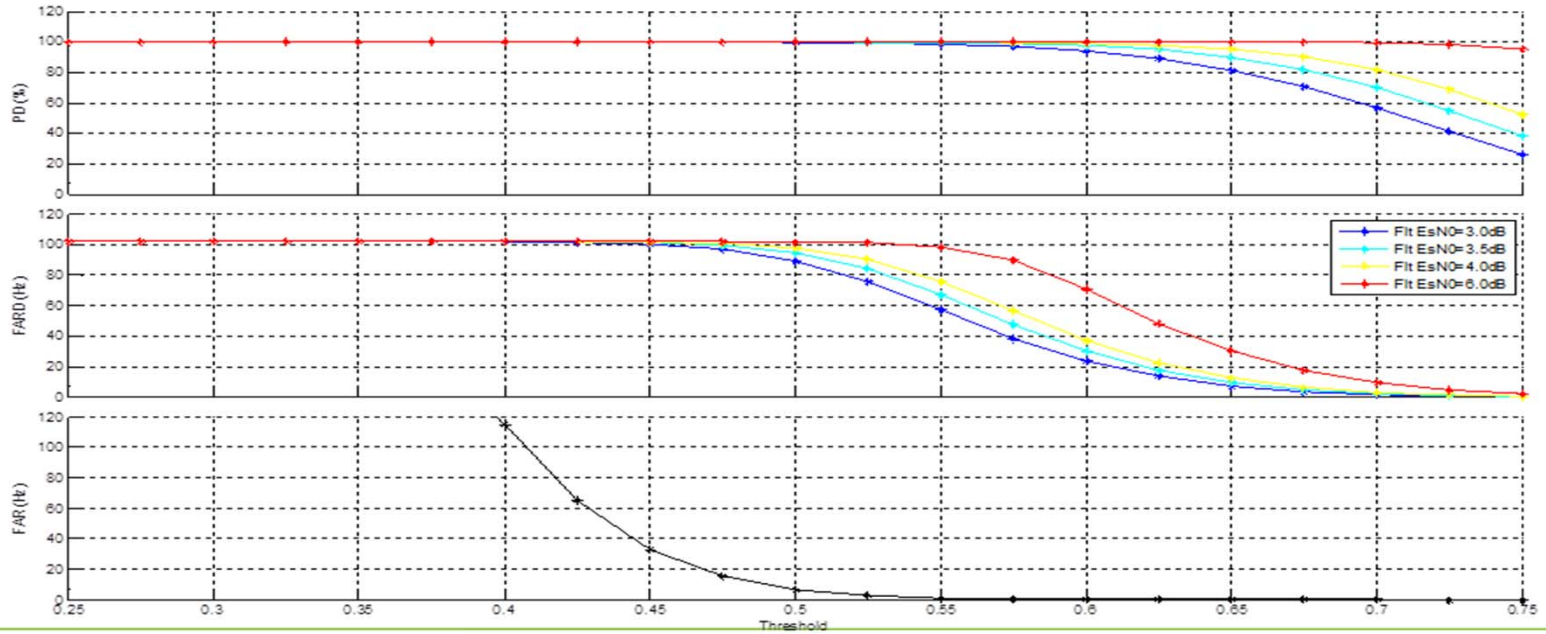
REAL-WORLD EXAMPLES
GMR-1 RELEASE 2 BURST DETECTOR #1



- Simultaneously channelise all possible channel positions from the entire 150MHz band (exact tuning and rates required)
  - 4800 x 31.25kHz ('1x rate' bursts)
  - 1920 x 125kHz ('4x rate' bursts)
  - 960 x 156.25kHz ('5x rate' bursts)
- Use continuous cross-correlation on all channels to search for the presence of expected bursts (using unique header symbols)
- Report detected bursts' time, channel and symbol rate

REAL-WORLD EXAMPLES
GMR-1 RELEASE 2 BURST DETECTOR #2

- All channelisation and detection functionality fits on a single Xilinx Virtex-6 FPGA (Pentek Cobalt board)
- QPSK detection @3.5dB EsN0:
  - Probability of Detection > 99.90% for False Alarm Rate of 1Hz
  - Probability of Detection > 99.99% for False Alarm Rate of 50Hz



SUMMARY

- RFEL provides proven FPGA-based DDCs and DUCs
- Several efficient architectures are available that provide trade-offs between:
  - Channel density
  - Bandwidth control
  - Sample rate control
  - Flexibility
- RFEL's CCF provides:
  - Build-time configurable architecture
  - Run-time configurable channel parameters
  - O (K log K) processing load

SIGNAL PROCESSING **IQ** : **FPGA** EXPERTISE

QUESTIONS?

CONTACT

Eddy Fry
Technical Director

T. +44 (0)1983 216628
M. +44 (0)7795 564569
F. +44 (0)1983 216611
E. eddy.fry@rfel.com

Unit B, The Apex
St Cross Business Park
Newport, Isle of Wight
PO30 5XW, UK