RF Engines Limited

"TPFT - Tuneable Pipelined Frequency Transform"

White Paper

Reference : W02003-Tuneable PFT White Paper
Revision : 1.0
Date : 19/09/2002
Author : A.Russo

Tel : +44 (0)1983 550330
Fax : +44 (0)1983 550340
E-mail : info@rfel.com
Web : www.rfel.com

RF Engines, Innovation Centre, St Cross Business Park, Newport, Isle of Wight, England, PO30 5WB

Table of contents

List of Figures

Introduction

Building upon the Pipelined Frequency Transform (PFT[1]) architecture, a new filter bank technique has been developed, which allows asymmetrical bands splitting of a given spectrum of frequencies.

A few applications require that an input signal be separated into a number of different frequency channels. If these channels are of equal width and equally spaced, then techniques such as the Fast Fourier Transform (FFT) or our proprietary Pipelined Frequency Transform (PFT) can be employed. For the general case of frequency bands of different width and unevenly distributed across the spectrum, then the most common solution is to employ a number of Digital Down Converters (DDC), each responsible for an individual channel.

The Tuneable Pipelined Frequency Transform (TPFT) provides similar functionality to a stack of DDCs. It gives the user freedom to specify channels by centre frequency and bandwidth, define filter characteristics and reconfigure to another frequency plan as required. Furthermore, spectral shaping masks can also be directly applied onto the outputs within the architecture itself.

This paper will provide a description of the TPFT architecture and will highlight the advantages of this technique over competing solutions.

General Description of Tuneable PFT (TPFT)

The pipelined architecture employed by the PFT serves well for the purpose of extracting different size frequency bands: at each PFT stage, the spectrum is separated into bands which are half as wide as those from the previous stage, and positioned as illustrated in figure 1.

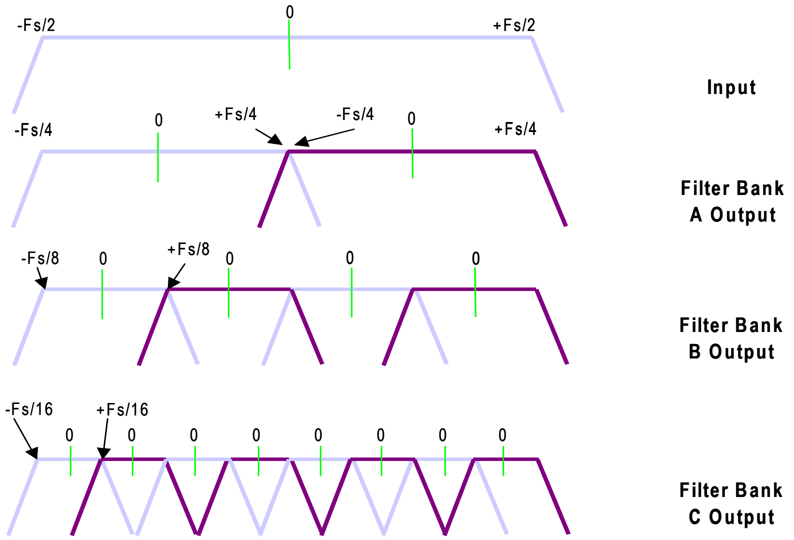
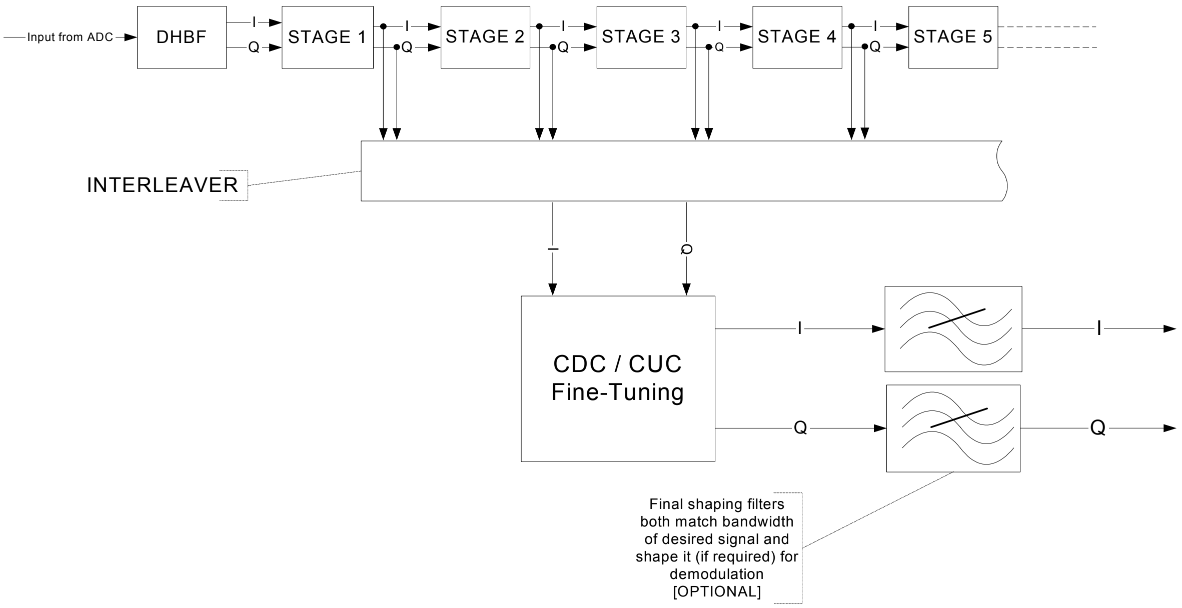


Figure 1 – Frequency separation of a standard PFT.

---

[1] The reader is strongly encouraged to read the PFT White Paper before continuing with this document. This can be downloaded from our web site at: http://www.rfel.com



Final shaping filters
both match bandwidth
of desired signal and
shape it (if required) for
demodulation
[OPTIONAL]

Figure 2 – Schematic view of TPFT typical architecture.

Due to the PFT cascaded architecture (figure 2), intermediate outputs are readily available. It is possible, by means of modifying the PFT architecture, not only to extract frequency bands of the desired size, but also to ensure these bands are centred at any given frequency.

This level of *tuneability* is achieved in two stages: firstly the signals are coarsely tuned within the PFT stages, then fine tuned by a complex converter whose Local Oscillator (LO) is a Numerically Controlled Oscillator (NCO) driven by the routing engine (a schematic of these subsystems is shown in figure 3 and 4).

The approach chosen here is to provide coarse tuning within the PFT stages, doing so by means of LO values chosen from the following set:

$\left\langle \frac{Fx}{16};\frac{Fx}{8};\frac{3Fx}{16} \right\rangle$ where $Fx = \frac{SampleRate}{2^{stage\_no}}$, i.e. the band rate.

Wider filters are used to ensure that the entire spectrum is covered and carried forward to the next stages. The filter passband has been extended by 50%, whist maintaining the same stopband characteristics.

It should be observed that the wider filter permits to include the same frequencies as the standard PFT, independently of the LO value used, with the added benefit of ensuring the centre of any carrier within the bin would not be further than Fx/16 from the centre of the bin itself.

The main advantage of performing the tuning operation in two steps is the reduction of size, for a given frequency resolution, of the LUT used for fine tuning: the fine-tuning mixing process only needs to shift frequencies by a maximum of $Fs_{bin}/16$ Hz instead of the full $Fs_{bin}/4$ Hz. This in practice translates to a fourfold LUT size saving for a given frequency resolution.

The benefit of such a reduction in LUT size can be best appreciated with a numerical example. Assume both a DDC and the TPFT have to extract a 100Khz channel from an input bandwidth of ~80 MHz, with a frequency resolution of 10Hz. The size of the

LUT used in the DDC NCO would be $8e^6$, while a 10-stage TPFT would achieve this resolution with a $1.25e^3$ values LUT. The reduction in LUT size is $\sim 1e^4$ orders of magnitude.

It should be noted that the frequency resolution is relative to the stage considered and is given by:

$Fx_{resolution} = (Fx/16) / LUT_{size}$, where Fx is the sample rate of every bin out of stage *x*.

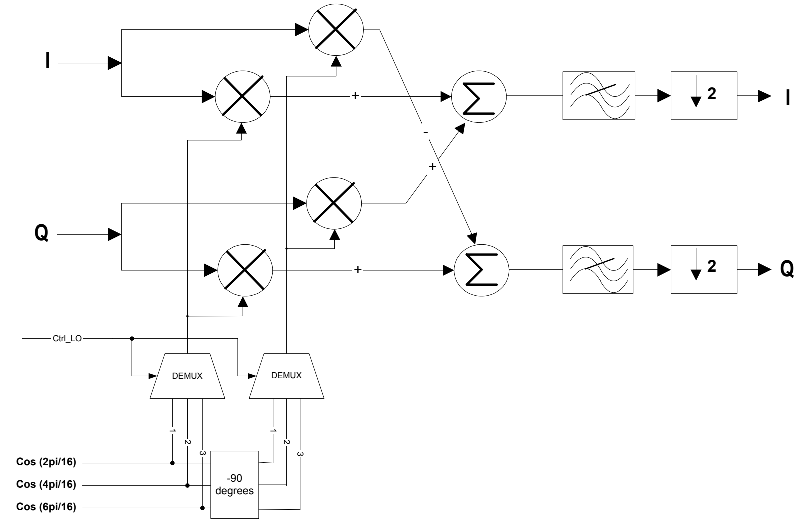


Figure 3 – Schematic view of Complex Down Converter for TPFT.

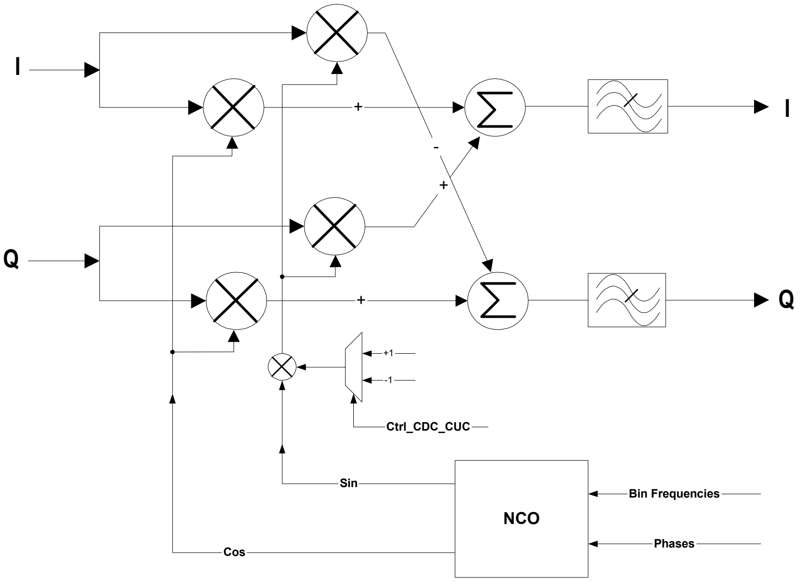


Figure 4 – Schematic view of fine tuning complex frequency converter used in the TPFT.

After fine tuning takes place, a polyphase[2] Finite Impulse Response (FIR) filter is used to extract only the required bandwidth for each signal (figure 5). This final filtering stage can also be used for independent spectral shaping/masking of each output bin.

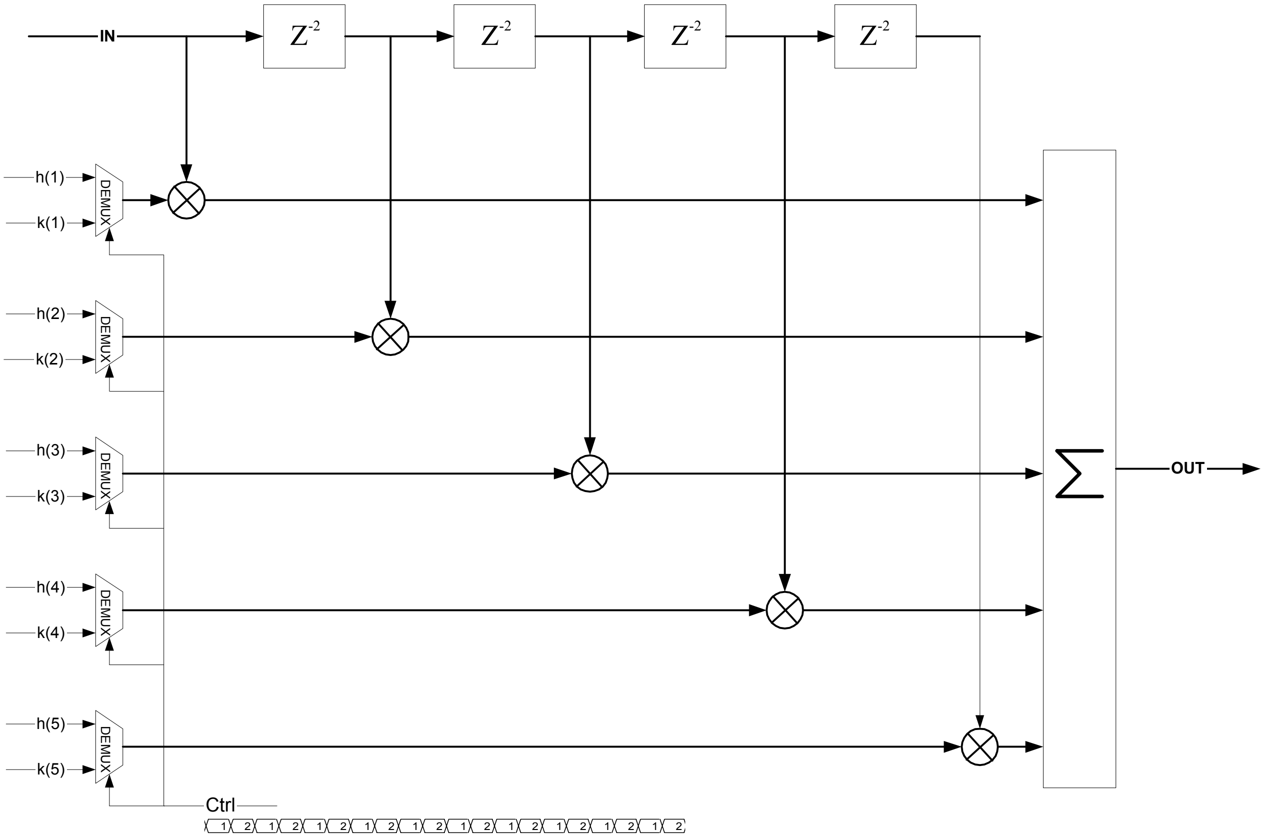


Figure 5 – Example of polyphase shaping filter for 2 channels shaping: one channel is filtered by filter *h(n)*, the other by filter *k(n)*.

As the sample rate of every band out of the TPFT is a power of two fraction of the input sample rate, it is possible to interleave all outputs from different stages within one output stream running at the full system rate. Furthermore, as long as the sum of all bands sample rate does not exceed the system rate, all of these can be accommodated in one output stream. For those familiar with the wavelet transform, the samples are interleaved in a similar manner to the one used for the output of discrete wavelet transform.

Previous to interleaving, every stage's output stream is buffered in a circular manner, thus all bins output are available at any given time. For each stage buffer, an address table as well as a counter are kept, so the correct bin can be extracted when its time slot on the interleaved output stream becomes available. The order by which intermediate stages are interleaved onto the output stream is stored together with

---

[2] A polyphase filter is a filter whose tap coefficients are interleaved in time, i.e. they change for every sample phase. Such structure is very useful for filtering interleaved data streams.

the information required for accessing the correct sample within each of the stages buffers. A schematic representation of this subsystem is shown in figure 6.

It should be noted that the interleaved output sample streams are not necessarily at the exact sampling rate for baseband demodulation. For different bandwidth channels that follow a similar bandwidth pattern (e.g. power of two step size), this can be addressed by adjusting the system rate. The general case requires a multi-rate section operating on each channel independently, and a solution for this is currently being studied.

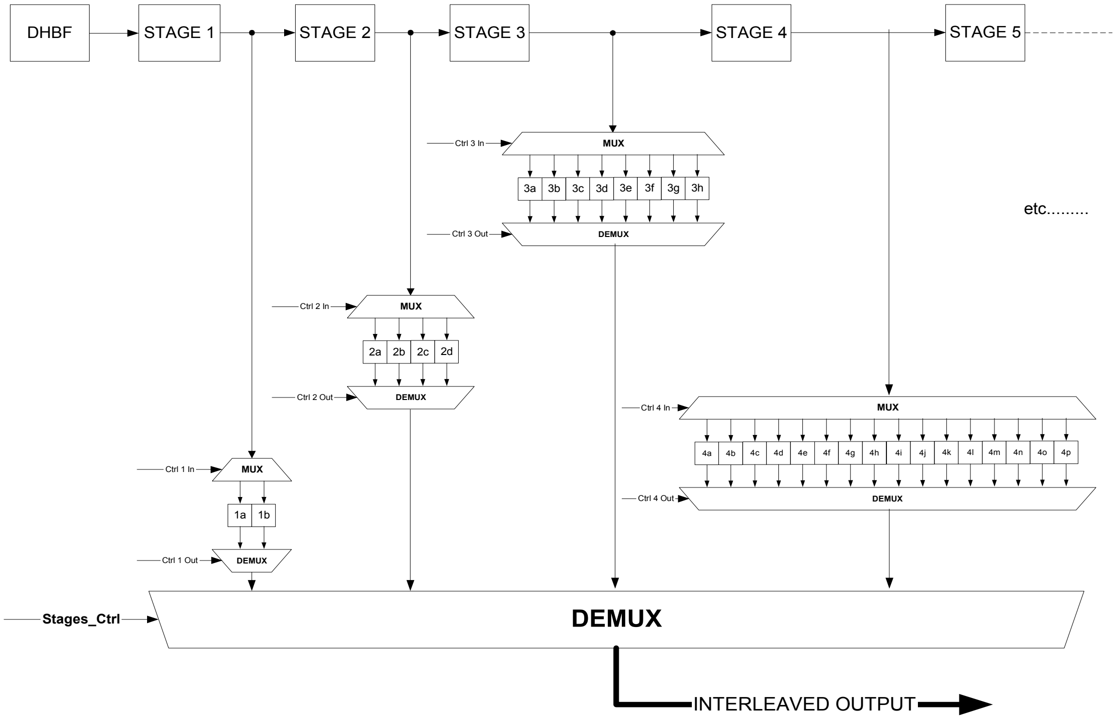


Figure 6 – Schematic view of output interleaver architecture.

Requirements for Tuneability

There are frequency splitting plans that require the fine tuning components of the TPFT to be duplicated. Typically, this happens when a large number of desired frequency spans are just about large enough for extraction within a given stage. As this implies that the effective sample rate for each of those bins is greater than twice oversampled, either or both of the following will happen:

- the output stream rate is insufficient for all channels required;
- some of the adjacent carriers becomes unreachable.

An example of such situation occurring is given in figure 7.

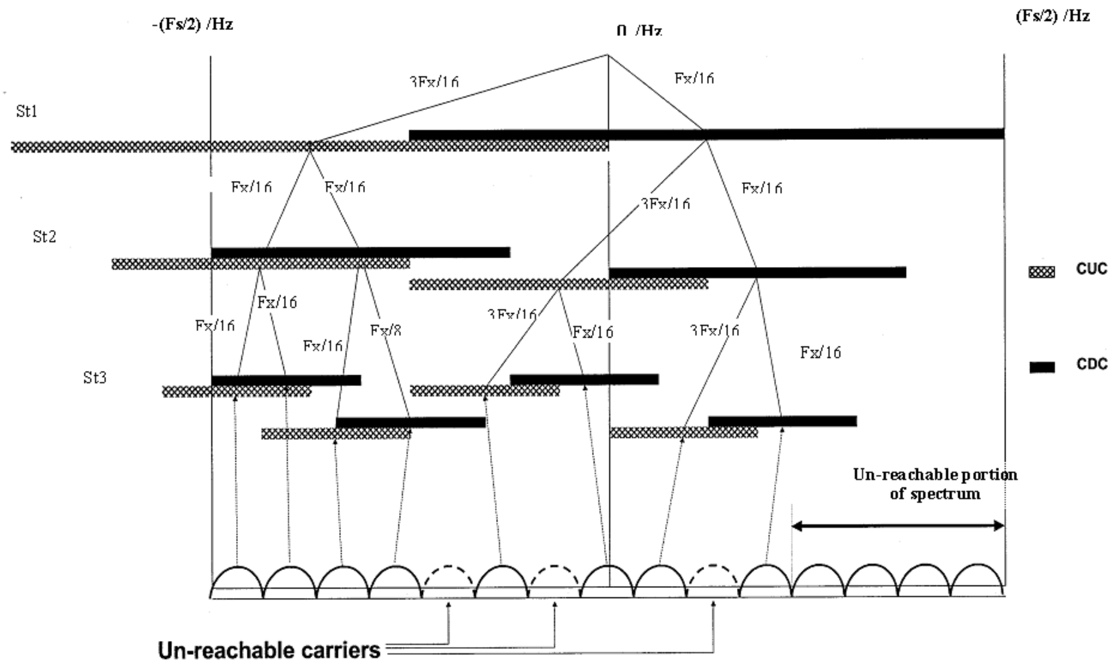


Figure 7 – Example: closely packed carriers of width slightly greater than the minimum required for extraction out of stage 3. The net result is that some of these carriers will not be extracted and a contiguous portion of the spectrum is completely out of reach.

In order to deal with this problem, two adjacent carriers of total bandwidth Bw (where Fx<Bw<1.5Fx) are extracted from a given bin. The issue of using more bandwidth than the one normally available in one bin is overcome by fine tuning both carries separately, thus resulting in a two output streams both running at the bin sample rate. A schematic view of such sub-system is shown in figure 8.

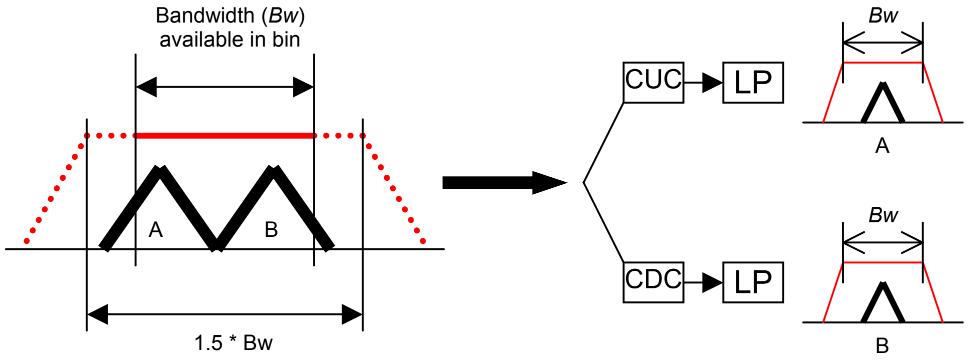


Figure 8 – Illustration of two carriers extraction from one bin.

In those situations where no assumption can be made on the channels required for extraction, the fine tuning components need to be duplicated, as illustrated in figure 9.

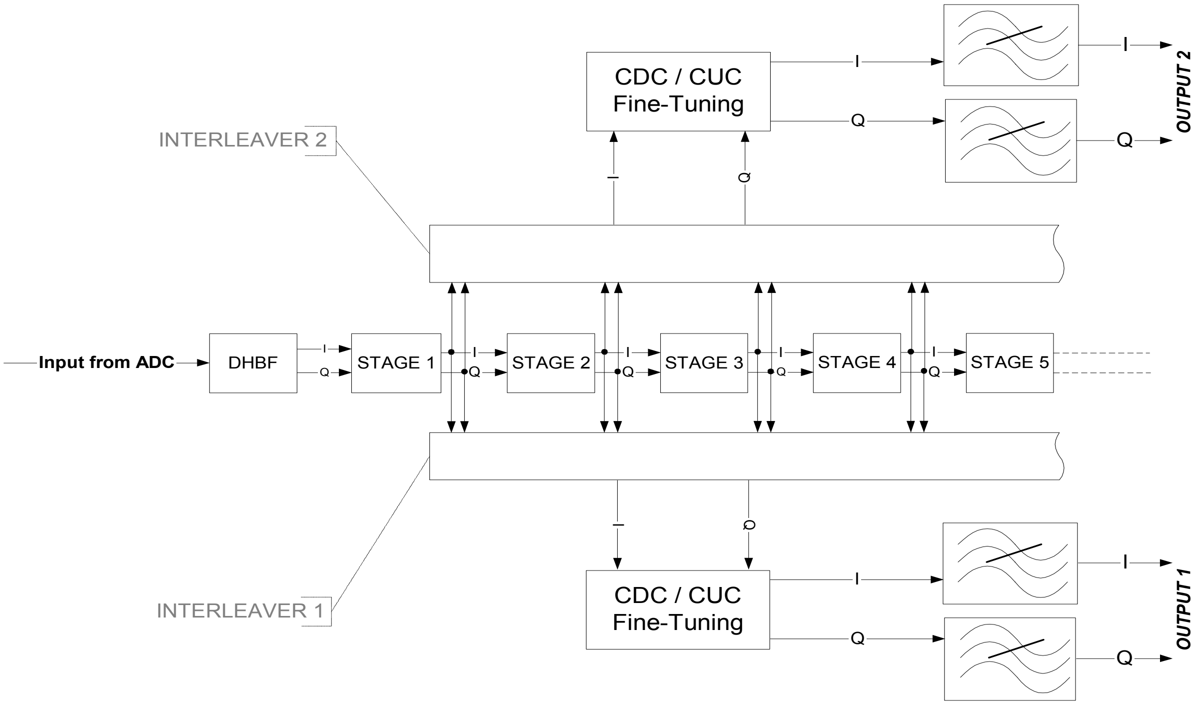


Figure 9 – Schematic of fully customisable TPFT.

However, the situation just illustrated is perhaps extreme: in most real life situations guard bands separate adjacent carriers. Furthermore, the first and last 10~15% of the spectrum are typically unusable: any anti-aliasing filter at the front-end of the PFT would need to have zero transition band to make 100% of the spectrum usable, or, alternatively, for the given bandwidth, the sample rate would need to be increased by a factor of 120%.

The trade-off between hardware complexity (single or double fine tuning subsystem) and flexibility in terms of guaranteeing full spectrum coverage for any given set of signals is application dependent and needs to be assessed on a case by case basis.

A final decision as to whether the use of two output streams is required, needs to be based on the balance between reducing hardware complexity and the typical bandwidth occupancy of the application in which the TPFT is used.

Frequency Reconfiguration

Perhaps the most important feature of the TPFT is its ability to be reconfigured in real time[3] to output different user-defined frequency bands. The total output delay due to reconfiguration depends mainly on the filters used within each stage, as these need to fill up with new values before they output reliable information. As a guide, the delay due to reconfiguration effects for a 10 stages TPFT varies typically between 5,000 to 15,000 clock cycles (25µs to 75µs at a system clock rate of 204.8 MHz). In addition to this, the physical interface to the hardware must also be considered as this dictates the time for the new parameters to be updated into the registers.

---

[3] The intrinsic delay due to the latency of the TPFT cannot be avoided, and this is equivalent to the latency of the particular TPFT design.

Although it is possible to manually change the TPFT parameters to achieve tuneability onto a band, this process can be completely automated. An algorithm to perform automatic routing onto carriers of interest, as well as calculating other parameters necessary for the TPFT to function, has been implemented. The user is left with the sole task of selecting the frequency plan of interest and the type of shaping to be applied onto the output (figure 10).

Software Demonstrator

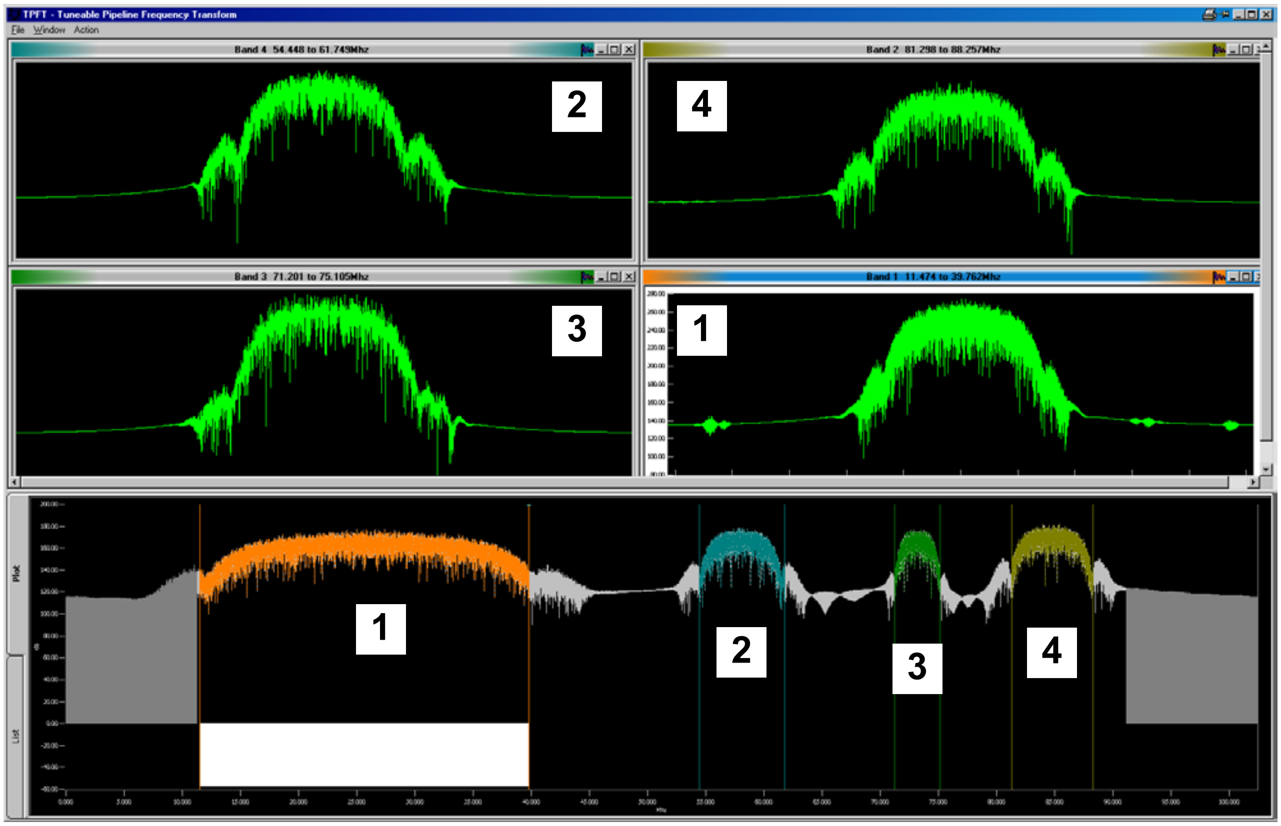


Figure 10 – Screen-shot of GUI software: 4 BPSK modulated signals of various bandwidth are extracted from one wideband input.

NOTE: flat passband shaping filters are used in this example, thus the signal's sidelobes fall within the filters' transition band and still show significant power spectral density above the noise floor.

An input signal is loaded from either an ASCII file or from our demonstrator board. Sampling rate and input bitwidth are then selected and the input spectrum shown on the bottom half of the screen. The two grey shadowed opposite ends of the spectrum represent the aliased frequencies, which the TPFT cannot operate upon. It should be noted that the amount of aliased spectrum is user defined (the default value is ~10% of the total bandwidth) and can be modified to represent any front-end filter roll-off.

The frequency bands of interest can then be directly selected on the spectrum plot itself or by entering their coordinates. Finally, different shaping filters can be applied onto each individual channel.

At this point the model can be started: on the basis of the channels selected, the number of necessary stages is determined and the TPFT tree is automatically routed, following which the input stream is passed through the TPFT itself. The outputs samples of each channel are stored and their spectrum displayed on screen.

System Architecture

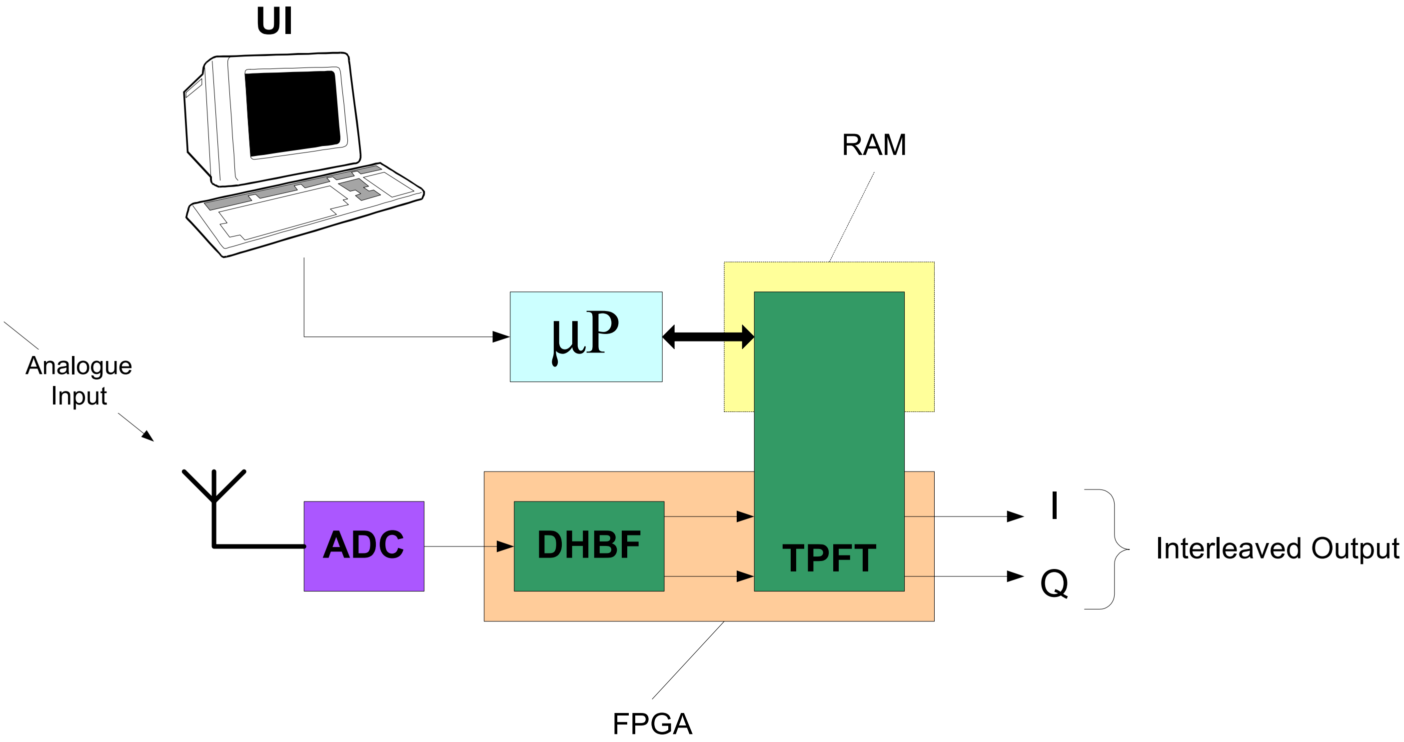


Figure 11 – Schematic view of a complete TPFT system. The µ-processor to configure the architecture to a frequency plan can either be a PC or a dedicated solution.

Figure 11 provides a schematic view of the main building blocks of a TPFT system. A graphical interface, as shown in figure 10, provides full controls for the engine. This is to be provided by RFEL, thus eliminating the need to generate a custom interface to the hardware, which in turn minimise risks and costs for a customer. The processing of the routing of the TPFT onto a specified frequency plan can be performed either on a general purpose PC, or can be incorporated onto the hardware. The latter, for instance, can be performed via any of the System-On Programmable Chip (SOPC) commercially available (e.g Xilinx® Virtex™-II Pro). The TPFT engine itself can be easily accommodated within a modern generation FPGA device such as a Xilinx® Virtex™-II or an Altera® Stratix™, as well as being developed as an ASIC. Additional memory might be required for larger transforms, mainly due to the increase in memory bandwidth needs of later stages. Finally, a Digital Half-Band Filter (DHBF) might also be incorporated to perform real-to-complex conversion of an already digitised signal.

Comparison of the TPFT with Conventional Digital Down-Converters

Standard digital down-converters (DDC) are probably the most common solution in those applications where only a limited number of channels are required. However, as the number of channels increases it becomes apparent that the linear growth in components count experienced by a stack of DDCs can be outperformed by an interleaving structure such as the TPFT. The growth in logic components count due to increased number of channels within a TPFT is proportional to $\log_2$, thus there would be a threshold point after which DDCs are inefficient when compared to the TPFT (figure 12). In reality, the TPFT memory requirements increase exponentially with the number of stage, until the point where the memory becomes the principal constraint. It is envisaged that as many as 16 stages (i.e. upto 65536 different channels) can be implemented using external memory to the FPGA or ASIC solution.

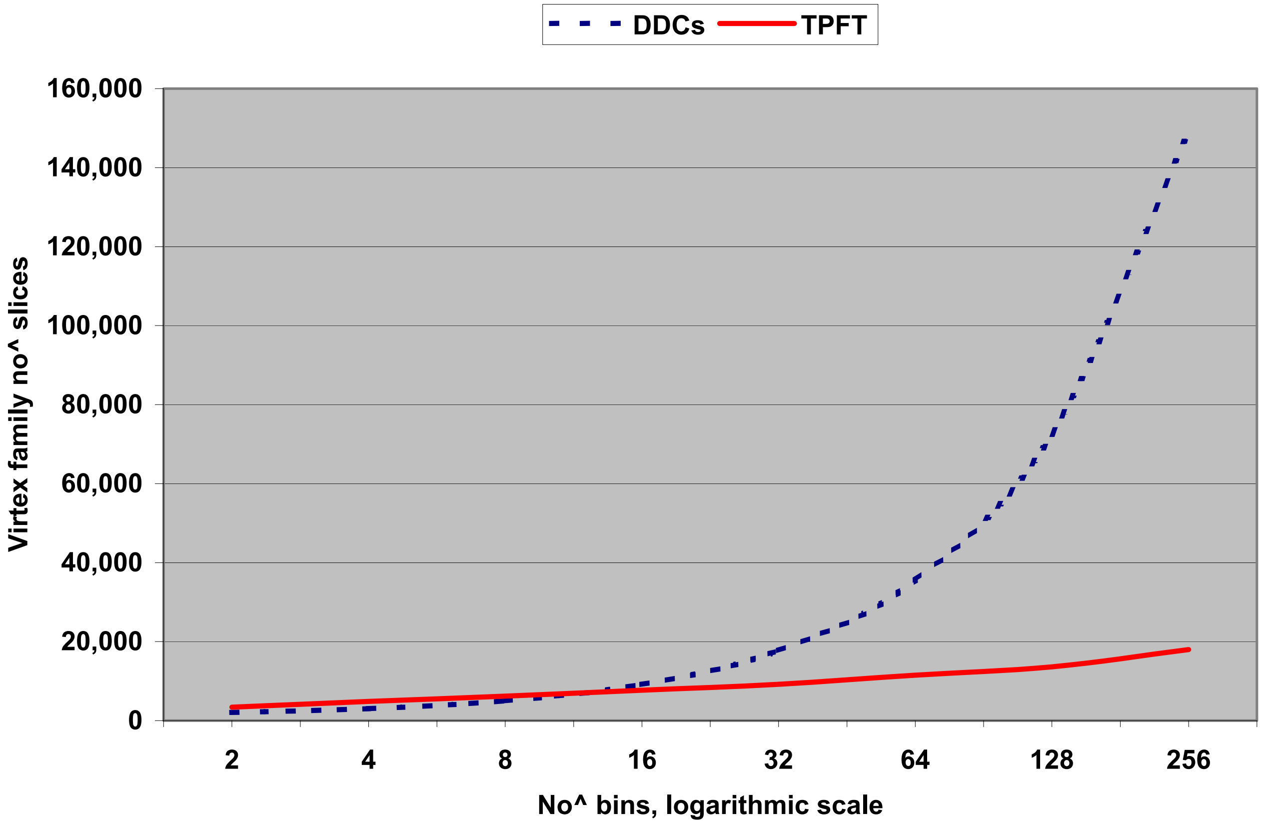


Figure 12 – Amount of logic required by DDC/TPFT for a given number of channels.

As an example, a TPFT design could extract 25 channels each ~1MHz in bandwidth from a wideband input of 100MHz with a 6-stage engine. This, depending on the filter requirements, would typically fit on a Xilinx Virtex-II 3000.

TPFT Application Example

In satellite communications, the allocation of frequencies within the available spectrum will change with the bandwidth requirements of the various operators. It is possible that at a particular time the channel configuration of figure 13a is being used. As the operators' requirements change, the new configuration of figure 13b is now employed. The use of the TPFT allows switching on the new frequency plan with minimal delays and without changes in the hardware. If a stack of DDCs were used instead, the reconfiguration process would still present the typical delay due to filter settling time, yet it would now require 10 more DDCs. As the number of channels grows, the use of DDC becomes more and more impractical when compared to the TPFT.

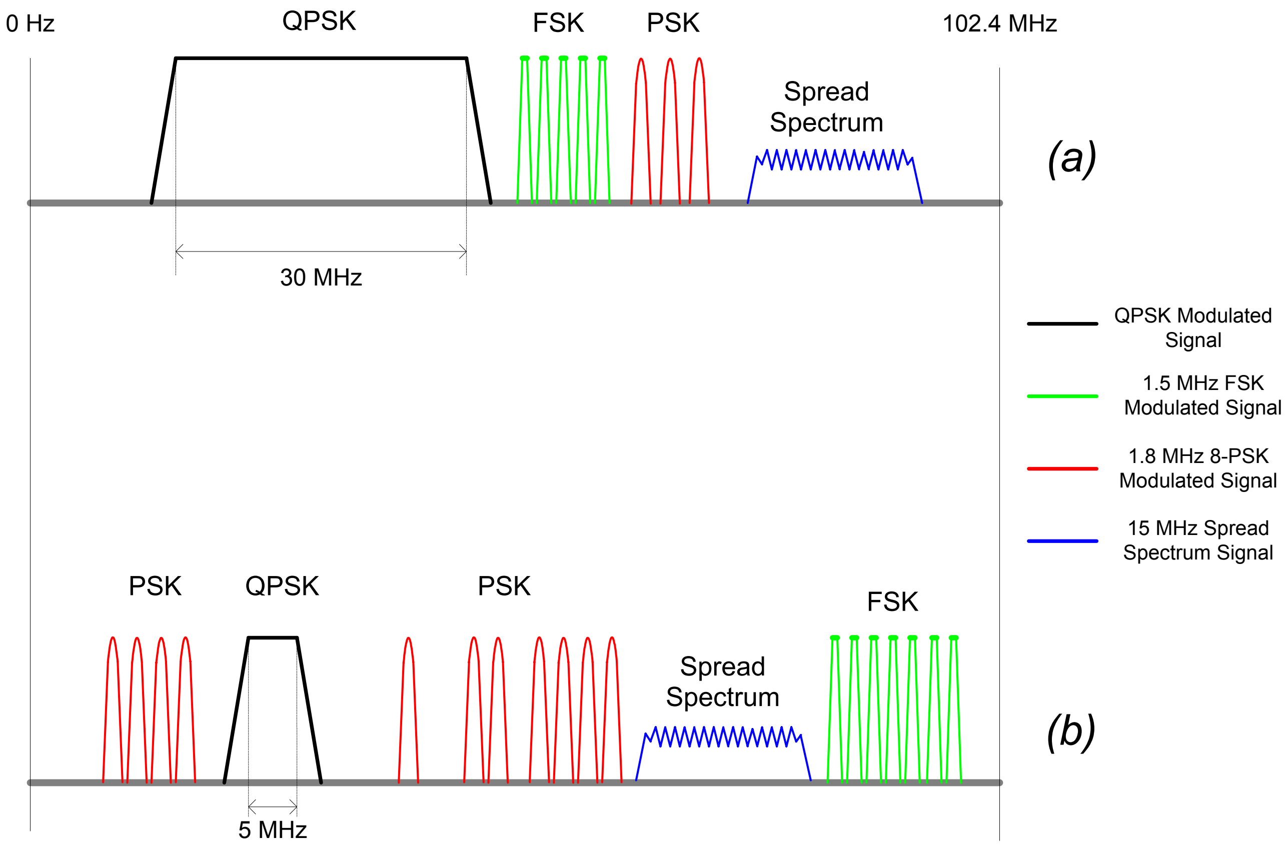


Figure 13 – Two possible frequency plans for a wideband satellite radio link.

Conclusions

Hopefully, this paper has demonstrated that for a plurality of simultaneous channels, the TPFT architecture provides significant advantages over more conventional methods of frequency splitting, such as complex down converters, or even less flexible frequency transforms such as the Discrete Fourier Transform (DFT), etc.

Overall, the TPFT fills the gap between the comparatively inflexible FFT or PFT approaches and the use of DDCs, which is extremely flexible but becomes increasingly inefficient above a certain number of channels. The TPFT provides a highly flexible and efficient means of frequency channelisation, which is fully re-configurable within its hardware frame.