

Real-time Nyquist signaling with dynamic precision and flexible non-integer oversampling

R. Schmogrow,^{1,6,*} M. Meyer,¹ P.C. Schindler,¹ B. Nebendahl,³ M. Dreschmann,²
J. Meyer,² A. Josten,¹ D. Hillerkuss,⁵ S. Ben-Ezra,⁴ J. Becker,² C. Koos,¹ W. Freude,¹
and J. Leuthold,^{1,5,7}

¹Institute of Photonics and Quantum Electronics (IPQ), Karlsruhe Institute of Technology (KIT), Germany

²Institute for Information Processing Technologies (ITIV), Karlsruhe Institute of Technology (KIT), Germany

³Agilent Technologies, Boeblingen, Germany

⁴Finisar, Nes Ziona, Israel

⁵ETH Zurich, Institute of Electromagnetic Fields (IFH), Zurich, Switzerland

⁶Now with: Infinera Corporation, Sunnyvale, California, USA

⁷JuergLeuthold@ethz.ch

*RSchmogrow@infinera.com

Abstract: We demonstrate two efficient processing techniques for Nyquist signals, namely computation of signals using dynamic precision as well as arbitrary rational oversampling factors. With these techniques along with massively parallel processing it becomes possible to generate and receive high data rate Nyquist signals with flexible symbol rates and bandwidths, a feature which is highly desirable for novel flexgrid networks. We achieved maximum bit rates of 252 Gbit/s in real-time.

©2016 Optical Society of America

OCIS codes: (060.0060) Fiber optics and optical communications; (060.1660) Coherent communications; (060.4080) Modulation.

References and links

1. G. Bosco, V. Curri, A. Carena, P. Poggiolini, and F. Forghieri, "On the performance of Nyquist-WDM terabit superchannels based on PM-BPSK, PM-QPSK, PM-8QAM or PM-16QAM subcarriers," *J. Lightwave Technol.* **29**(1), 53–61 (2011).
2. W. Shieh, H. Bao, and Y. Tang, "Coherent optical OFDM: theory and design," *Opt. Express* **16**(2), 841–859 (2008).
3. R. Schmogrow, M. Winter, M. Meyer, D. Hillerkuss, S. Wolf, B. Baeuerle, A. Ludwig, B. Nebendahl, S. Ben-Ezra, J. Meyer, M. Dreschmann, M. Huebner, J. Becker, C. Koos, W. Freude, and J. Leuthold, "Real-time Nyquist pulse generation beyond 100 Gbit/s and its relation to OFDM," *Opt. Express* **20**(1), 317–337 (2012).
4. R. Schmogrow, M. Winter, D. Hillerkuss, B. Nebendahl, S. Ben-Ezra, J. Meyer, M. Dreschmann, M. Huebner, J. Becker, C. Koos, W. Freude, and J. Leuthold, "Real-time OFDM transmitter beyond 100 Gbit/s," *Opt. Express* **19**(13), 12740–12749 (2011).
5. R. Bouziane, R. Schmogrow, D. Hillerkuss, P. A. Milder, C. Koos, W. Freude, J. Leuthold, P. Bayvel, and R. I. Killey, "Generation and transmission of 85.4 Gb/s real-time 16QAM coherent optical OFDM signals over 400 km SSMF with preamble-less reception," *Opt. Express* **20**(19), 21612–21617 (2012).
6. B. Inan, S. Adhikari, O. Karakaya, P. Kainzmaier, M. Mocker, H. von Kirchbauer, N. Hanik, and S. L. Jansen, "Real-time 93.8-Gb/s polarization-multiplexed OFDM transmitter with 1024-point IFFT," *Opt. Express* **19**(26), B64–B68 (2011).
7. D. Qian, T. Kwok, N. Cvijetic, J. Hu, and T. Wang, "41.25 Gb/s real-time OFDM receiver for variable rate WDM-OFDMA-PON transmission," in *Optical Fiber Communication Conference*, OSA Technical Digest (CD) (Optical Society of America, 2010), paper PDPD9.
8. N. Kaneda, Q. Yang, X. Liu, S. Chandrasekhar, W. Shieh, and Y. Chen, "Real-time 2.5 GS/s coherent optical receiver for 53.3-Gb/s sub-banded OFDM," *J. Lightwave Technol.* **28**(4), 494–501 (2010).
9. M. Yoshida, T. Omiya, K. Kasai, and M. Nakazawa, "Real-time FPGA-based coherent optical receiver for 1 Gsymbol/s, 64 QAM transmission," in *Optical Fiber Communication Conference*, OSA Technical Digest (CD) (Optical Society of America, 2011), paper OTuN3.
10. R. Schmogrow, R. Bouziane, M. Meyer, P. A. Milder, P. C. Schindler, R. I. Killey, P. Bayvel, C. Koos, W. Freude, and J. Leuthold, "Real-time OFDM or Nyquist pulse generation--which performs better with limited resources?" *Opt. Express* **20**(26), B543–B551 (2012).
11. H. Nyquist, "Certain topics in telegraph transmission theory," *Trans. Am. Inst. Electr. Eng.* **47**(2), 617–644 (1928).

12. S. D. Personick, "Receiver design for digital fiber optic communication systems, I," *Bell Syst. Tech. J.* **52**(6), 843–874 (1973).
13. S. Smith, *The Scientist & Engineer's Guide to Digital Signal Processing*, (California Technical Pub., 1997), p. 45, <http://www.dspguide.com/>.
14. R. Schmogrow, M. Meyer, S. Wolf, B. Nebendahl, D. Hillerkuss, B. Baeuerle, M. Dreschmann, J. Meyer, M. Huebner, J. Becker, C. Koos, W. Freude, and J. Leuthold, "150 Gbit/s Real-time Nyquist pulse transmission over 150 km SSMF enhanced by DSP with dynamic precision," in *Optical Fiber Communication Conference*, OSA Technical Digest (Optical Society of America, 2012), paper OM2A.6.
15. R. Schmogrow, M. Meyer, P. C. Schindler, A. Josten, S. Ben-Ezra, C. Koos, W. Freude, and J. Leuthold, "252 Gbit/s real-time Nyquist pulse generation by reducing the oversampling factor to 1.33," in *Optical Fiber Communication Conference*, OSA Technical Digest (Optical Society of America, 2013), paper OTu2I.1.
16. R. Schmogrow, B. Nebendahl, M. Winter, A. Josten, D. Hillerkuss, S. Koenig, J. Meyer, M. Dreschmann, M. Huebner, C. Koos, J. Becker, W. Freude, and J. Leuthold, "Error vector magnitude as a performance measure for advanced modulation formats," *IEEE Photon. Technol. Lett.* **24**(1), 61–63 (2012).
17. R. Schmogrow, B. Nebendahl, M. Winter, A. Josten, D. Hillerkuss, S. Koenig, J. Meyer, M. Dreschmann, M. Huebner, C. Koos, J. Becker, W. Freude, and J. Leuthold, "Corrections to: Error vector magnitude as a performance measure for advanced modulation formats: erratum," *IEEE Photon. Technol. Lett.* **24**(23), 2198 (2012).
18. R. Schmogrow, S. Ben-Ezra, P. Schindler, B. Nebendahl, C. Koos, W. Freude, and J. Leuthold, "Pulse-Shaping With Digital, Electrical, and Optical Filters – A Comparison," *J. Lightwave Technol.* **31**(15), 2570–2577 (2013).
19. A. Nespola, S. Straullu, G. Bosco, A. Carena, and P. Yanchao Jiang, Poggiolini, F. Forghieri, Y. Yamamoto, M. Hirano, T. Sasaki, J. Bauwelinck, and K. Verheyen, "1306-km 20×124.8-Gb/s PM-64QAM transmission over PSCF with net SEDP 11,300 (bkm)/s/Hz using 1.15 samp/symb DAC," in *39th European Conference and Exhibition on Optical Communication (ECOC 2013)*, paper Th.2.D.1.
20. T. Mizuochi, "Recent progress in forward error correction and its interplay with transmission impairments," *IEEE J. Sel. Top. Quantum Electron.* **12**(4), 544–554 (2006).

1. Introduction

Next generation networks rely on real-time transmitters (Tx) and receivers (Rx) that allow the generation and reception of phase and amplitude modulated signals in combination with spectrally efficient pulse-shapes. However, real-time processing of multi-gigabit data streams is highly demanding and only practical if algorithms are found that provide high computational precision with little processing effort. In addition, symbol rates should be as close as possible to the sampling rates offered by state-of-the-art digital-to-analog and analog-to-digital converters (DAC and ADC) in order to efficiently make use of the available speed on the one hand and for minimum energy consumption on the other hand. Last but not least, developed techniques should be compatible with emerging novel flexgrid networks by offering dynamic adaptation of symbol rate and bandwidth.

In view of signals with optimum spectral efficiency (SE), there are two main contenders, namely Nyquist signaling [1] and orthogonal frequency division multiplexing (OFDM) [2]. Both pulse-shaping techniques offer a SE close to the theoretical limit. Real-time Tx and Rx for up to 101.5 Gbit/s (Tx) [3–6] and up to 41.25 Gbit/s (Rx) [7–9] have been already demonstrated. While these numbers are already impressive, the ultimate speed is limited by the specific implementation needs for either pulse-shaping technique. Recently, the limitations for digital real-time Nyquist and OFDM systems have been investigated [10]. While Nyquist signaling offers advantages such as higher out-of-band suppression [10] or a potentially lower peak-to-average power ratio [3], it also poses implementation challenges: Nyquist pulse-shaping requires a high computational precision due to the fast decay of the sinc-shaped impulse responses, and a large sampling rate with the usually used two-fold oversampling. Finally, processing of data rates beyond 100 Gbit/s can be only achieved using a high degree of parallelization, which is a challenge of its own.

Another prominent requirement for the application of digitally processed Nyquist and OFDM signals in flexgrid networks is the capability to change the symbol rate and thereby the signal bandwidth during runtime. In order to elaborate the challenges of optical flexgrid networks, Fig. 1 exemplarily shows a network with three frequency channels (f_0 red, f_1 black and f_2 blue). We illustrate how input data are processed by digital signal processing (DSP) and fed to two DACs. The figure shows spectra of signals at different positions in the Tx as

insets. For the signal measured at the DAC outputs, it can be seen that the spectra comprise both, the spectrum of interest (black) and also so-called image spectra (white) that repeat infinitely for an ideal DAC. The latter are removed by electrical low-pass filters (transfer function: dashed line). Nyquist sampling at the symbol rate would call for non-realizable “brick wall” filters, therefore some degree of oversampling by a factor q is required for accommodating real-world filter slopes in the guard bands (GB) between spectrum of interest and adjacent images. The oversampled data are then encoded on an optical carrier (laser diode, LD) with an optical modulator. Subsequently, the signals of several Tx (framed red, black, and blue) are multiplexed (MUX). The Rx then coherently receives one of the demultiplexed (DEMUX) channels using a 90° optical hybrid and photo detectors followed by another set of analog electrical filters, ADCs, and DSP.

The ultimate versatility is obtained when Tx and Rx can be flexibly adapted to optimally utilize the assigned bandwidth in the network (Fig. 1, Scenarios A and B). Unfortunately, the DSPs and filters do not allow for such flexible adaptation of the bandwidth by changing the sampling frequency. Designs for DSP circuits (e.g. field programmable gate arrays, FPGA, or application specific integrated circuits, ASIC) are typically built to be operated at a fixed frequency only. Any major change of the clock rate would require either a reprogramming or even a redesign. Likewise, the cut-off frequency of the analog image rejection (Tx) and antialias (Rx) filters is fixed by design and therefore removing any image spectra (Tx) or preventing aliasing (Rx) is done for one fixed sampling frequency only.

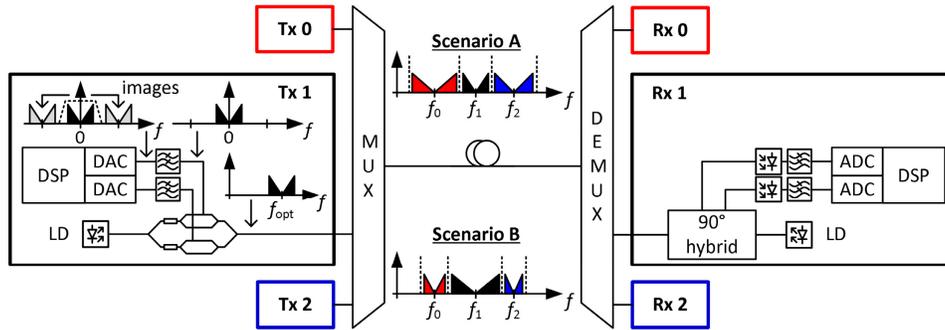


Fig. 1. Vision of a flexgrid optical network, where various Tx generate signals with adjustable bandwidths. In this example, the middle channel Tx (at frequency f_i) adapts its signal bandwidth flexibly. In Scenario A, Tx 1 generates signals with small bandwidth as its neighboring Tx 0 and Tx 2 (f_0 and f_2) occupy large bandwidths. In Scenario B, Tx 1 may use more bandwidth as its neighbors occupy small bandwidths. To realize networks with channels of flexible bandwidths, advanced algorithms for Tx and Rx digital signal processing (DSP) need to be found especially as the hardware does not support changing clock frequencies.

Alternative ways to change the bandwidth without changing the sampling frequency need to be found. This is possible by tuning the oversampling factor q rather than the sampling frequency (rate) f_s of the signal converters (DACs and ADCs). To understand this process, we take a closer look at oversampling. According to the Nyquist sampling theorem a bandwidth-limited signal can be fully represented by two real-valued or one complex sample per period of the highest frequency component in the signal [11]. This corresponds to one complex sample per symbol period, i.e., $q = 1$. In general, the relation between symbol rate F_s , sampling rate f_s , and oversampling factor q follows from:

$$\text{symbol rate } F_s = \frac{\text{sampling rate } f_s}{\text{oversampling factor } q}. \quad (1)$$

For Nyquist sinc-pulses, the signal bandwidth B equals the symbol rate $B = F_s$. For Nyquist signals comprising so-called “raised-cosine” pulses [12], the relation between symbol rate and

signal bandwidth is defined through the roll-off factor β and is described by $B = F_s (1 + \beta)$ for $0 \leq \beta \leq 1$. In general, changing the symbol rate F_s and thus the signal bandwidth B causes a change of the oversampling factor q if the sampling rate f_s is kept constant.

The effect of changing the symbol rate F_s (the bandwidth B) with constant sampling rate f_s is illustrated in Fig. 2. It shows both, Nyquist signals and OFDM signals, processed with different q but constant f_s . If signals are not oversampled, i.e., $q = 1$, then main spectrum and image spectra adjoin, see Fig. 2(a) and 2(b). Ideal “brick wall” analog filters would be needed to fully remove the image spectra without affecting the main spectrum for Nyquist signals, Fig. 2(a). For OFDM signals with $q = 1$ it is impossible to filter the images without affecting the main spectrum. In order to remove the image spectra with realizable filters, a guard band (GB) is needed. This is obtained for an oversampling factor $q > 1$, see Fig. 2(c)–2(f) for Nyquist and OFDM signals with $q = 2$, and $q = 4 / 3$, respectively [13]. In practice, an oversampling factor $q = 2$ is commonly used [3], because in this case the DSP implementation is straight-forward. It can be seen in Fig. 2(c)–2(f) that for a fixed sampling frequency f_s , the same analog filters as for $q = 2$ (schematic transfer function indicated by dashed lines) can be used to remove the image spectra. Therefore the signal bandwidth can be changed through the oversampling factor q , while the hardware and sampling frequency f_s is kept constant. In OFDM systems, the signal bandwidth and thus the oversampling factor q can be simply changed by nulling more or less of the SCs close to the edge of the main band. For Nyquist signals that are usually generated using finite-duration impulse response (FIR) filters, effectively changing the signal bandwidth through an adaptive oversampling factor q is more challenging.

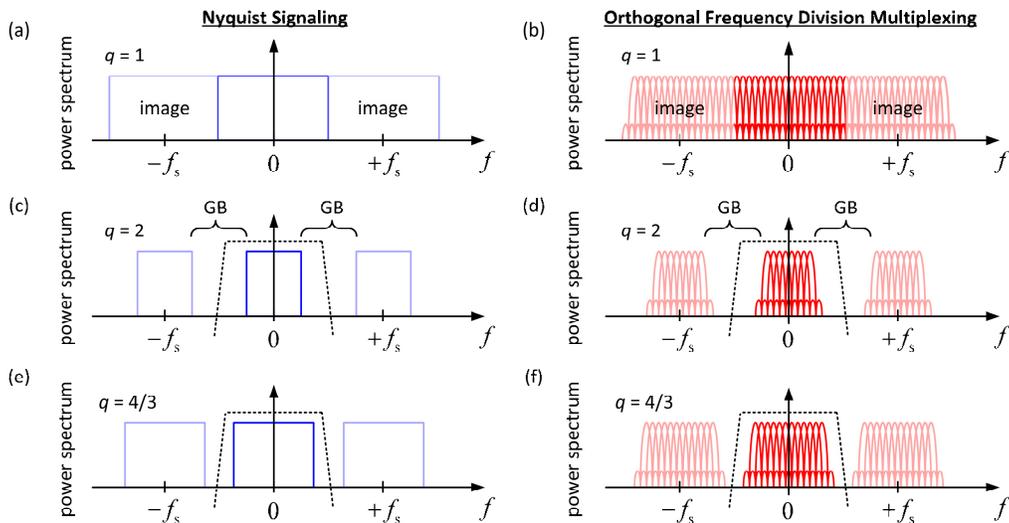


Fig. 2. Spectra of Nyquist and OFDM signals with oversampling factors $q = 1$ (no oversampling), $q = 2$ and $q = 4 / 3$. The main spectra are centered on zero frequency. Image spectra are displayed in light color at multiples of the sampling frequency f_s . These spectra are removed by analog filters (schematic transfer function indicated by dashed lines). (a) Spectrum of a Nyquist signal without oversampling. (b) OFDM spectrum ($q = 1$). (c) Electrical spectrum of a Nyquist signal ($q = 2$). The spectral guard band (GB) equals the electrical signal bandwidth. (d) OFDM spectrum ($q = 2$). (e) Nyquist spectrum with reduced oversampling factor $q = 4 / 3$. For a fixed sampling frequency f_s the symbol rate is increased while the GB is reduced. (f) Corresponding OFDM spectrum for $q = 4 / 3$.

In this paper, we first discuss a novel real-time sinc-pulse shaping algorithm that provides superior signal quality compared to conventional fixed-point arithmetic, while keeping the processing effort at minimum. This is obtained by dynamically changing the effective word length for computation [14]. As a result, we can decrease the bit error ratio (BER)

significantly without increasing the required computational resources. Second, we reduce the usual oversampling factor $q = 2$ to $q = 4 / 3 \approx 1.33$, thus increasing the symbol rate F_s at a constant sampling frequency f_s by 50% [15]. Third, we present an efficient parallel processing technique for generating and receiving Nyquist signals with adaptable oversampling factor q in order to process signals with variable symbol rates and bandwidths. All presented results are experimentally validated using a real-time Tx. To verify the advantages of processing with dynamical computational precision, we transmit polarization division multiplexed (PDM) 16QAM (64QAM) signals over up to 300 km (150 km) of standard single mode fiber (SSMF). To demonstrate Nyquist signals with a rational oversampling factor of $q \approx 1.33$, we transmit QPSK, 16QAM, and PDM-64QAM signals. A maximum bit rate of 252 Gbit/s (PDM-64QAM) is achieved for signals transmitted over up to 100 km of ultra-large area fiber (ULAF). Although showing the Tx only, the developed algorithms are also suitable for a potential real-time Rx.

2. Nyquist DSP with dynamically adjusted precision

Generating Nyquist signals with maximum SE calls for FIR filters of high order R [3]. However, increasing the filter order R (or in other words the number of filter coefficients) is problematic for sinc-pulses, since coefficients representing the filter's impulse response (IR) far away from the center peak of the IR show very small magnitudes.

On the one hand, high computational precision is needed to correctly represent small and large magnitudes of the IR at the same time. On the other hand, FPGAs and ASICs are the only reasonable choices for multi-gigabit processing, and they typically use fixed-point integer arithmetic so that the computational effort scales with the word length.

To provide a large effective word length without increasing the processing effort we dynamically adapt the computational precision within different intervals of the IR. To do so, an elementary sinc-impulse (with amplitude '1' at $t = 0$) is divided into intervals that share a multiplication factor 2^s , Fig. 3(a). All sampled floating-point (FP) values (word length 32 bit) of the sinc-impulse are scaled to a maximum signed value of $2^5 - 1$, which is chosen to match the final DAC resolution of 6 bit. The outer sinc-magnitudes (far away from $t = 0$) are close to zero. Therefore, each FP value is multiplied by 2^s , where the weight s is chosen — according to the sinc-magnitude — such that rounding to the closest 6 bit signed integer results in a maximum number of non-zero most significant bits.

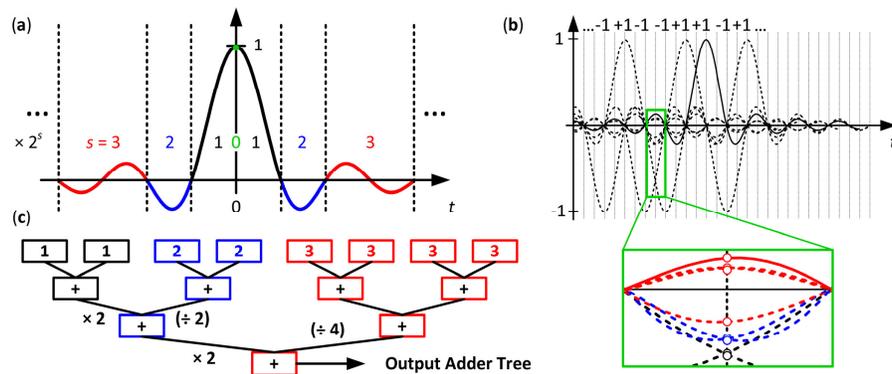


Fig. 3. DSP with dynamical increase of the computational word length. (a) An elementary sinc-impulse with amplitude 1 at $t = 0$ is scaled, multiplied with factors 2^s , and rounded to a signed 6 bit integer. The weight s is chosen according to the magnitude decrease of the sinc-function. (b) An output waveform resulting from the superposition of various sinc-impulses. Samples (see green-framed blow-up) for each point of time are added. (c) Adder tree for groups with different s uses either multiplications (\times , more accurate) or divisions (\div) to merge the groups.

The output signal is constructed from the superposition of modulated elementary sinc-impulses. A train of pulses is shown in Fig. 3(b). For simplicity, the modulation coefficients are chosen to be ± 1 . The output waveform is computed by adding the samples of all impulses belonging to the same point in time. The inset (framed green) shows 8 samples (\circ, \circ, \circ) to be added. An adder tree as seen in Fig. 3(c) groups samples with equal s and computes the output values, taking into account the previously assigned weights. When adding groups with unequal weights (e.g. $s_1 = 1$ and $s_2 = 2$ as in Fig. 3(c)), either the values of group s_1 must be divided by $2^{s_1-s_2}$ ($\times 2$ in Fig. 3(c)), or the values of group s_2 must be divided by $2^{s_2-s_1}$ ($\div 2$ in Fig. 3(c)). The (\times)-procedure results in better accuracy than the (\div)-method. This algorithm has to be applied for all adder stages and results in an improved effective word length of $p_{\text{eff}} = 9.9$ bit, instead of 7.9 bit for conventional, “standard” DSP.

3. Nyquist DSP with rational oversampling factor

Digital-to-analog converters and ADCs only offer sampling rates f_s up to a certain speed. This sets an upper limit to the achievable symbol rate F_s . If an oversampling factor $q = 2$ is used as in previous work [3], then the achievable symbol rate F_s is limited to $f_s / 2$. To overcome this limitation, we introduce a rational oversampling factor $q = k / l$ and discuss how such a filter can be implemented without computational overhead.

One way to generate sinc-pulses with an arbitrary rational $q > 1$ is to operate the FIR pulse-shaper at a clock rate $k F_s$ and to use every l -th sample for defining the output waveform while dropping all samples in between. This method, however, is unfavorable in terms of computational complexity, as samples have to be processed that do not contribute to the output. The same argument holds for the Rx, where $q = k / l$ oversampled signals need to be resampled by $q^{-1} = l / k$ so that at the output of the Rx FIR filter, exactly one sample per transmitted symbol, is obtained. In the following, we discuss a method to directly process q -fold oversampled sinc-pulses without overhead as will be explained for the example $q = 4 / 3$.

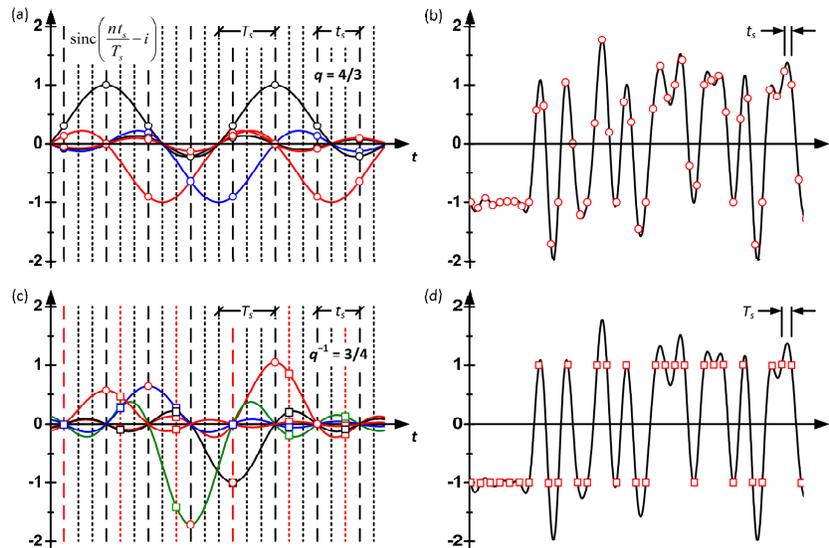


Fig. 4. Nyquist signaling with rational oversampling factor $q = k / l = 4 / 3$. The sampling period t_s is a fraction of T_s . (a) At the Tx, only every l -th sample is considered (open circles, dashed grid) instead of computing k samples per symbol (dotted grid). The IR is sampled at different relative sampling positions which repeat after $l = 3$ pulses (black, red, and blue). (b) Output waveform at the Tx. (c) At the Rx, the sampled signal (open circles) is interpolated with sinc-functions to recover the transmitted data. This time, only every k -th sample needs to be processed (open squares, red grid-lines). A number of $k = 4$ differently sampled sinc-functions (black, red, blue, and green) need to be provided as IR of the Rx filter. (d) Waveform after resampling. Data is recovered ISI-free (open squares).

The processing performed in the Tx is indicated in Fig. 4(a). It shows a number of sinc-pulses separated by the symbol period T_s . For the sake of visibility and without loss of generality, all sinc-pulses are modulated with real modulation coefficients ± 1 . Instead of computing all samples for ($k = 4$)-fold oversampling (indicated by the total of dotted and dashed grid lines), only samples located on the dashed grid (open circles) need to be summed to form the output waveform shown in Fig. 4(b). It should be noted, that since the sample period $t_s = 1 / f_s$ is a fraction of the symbol period T_s , we need to sample the IR at different relative positions. Fortunately, these relative sample positions repeat after l pulses so that only $l = 3$ differently sampled sinc-pulses (black, red and blue) are required for generating the output waveform in Fig. 4(b).

The processing performed in the Rx is indicated in Fig. 4(c). The sampled signal (open circles) is interpolated using sinc-functions. This is achieved by a similar FIR filter as has been used in the Tx. This time, however, the signal is resampled by a factor $q^{-1} = l / k = 3 / 4$. Instead of computing all samples lying on the depicted grid (dotted and dashed) in Fig. 4(c), only every k -th sample needs to be considered (open squares, red grid-lines). Therefore, just as it is done in the Tx, the sinc-shaped IR of the FIR filter is sampled at different relative positions which repeat after $k = 4$ pulses (black, red, blue, and green). Summing all samples (open squares) on this grid, the transmitted data (here: modulation coefficients ± 1) can be recovered without intersymbol interference (ISI). The outcome is depicted in Fig. 4(d) where the reconstructed waveform is shown.

The situation changes if neighboring Nyquist channels exist and the spectral guard band between adjacent channels is small. In this case, fragments of the neighbors are not removed by the analog electrical anti-alias filters preceding the ADCs, red-shaded spectral portions in Fig. 5(a). The appropriate Rx digital filtering for this scenario is visualized in Fig. 5(a). The analog signal is sampled by an ADC which, in this example, is operated at the same sampling rate as the Tx DACs. In general, the ADC's sampling rate may be chosen as low as 1 sample per symbol (SPS) without introducing any penalty as long as neighboring channels are fully removed in the analog domain. In this example, however, a digital representation of the signal with $4 / 3 \approx 1.33$ SPS is obtained. Looking at the spectrum to the right of the ADC (second from left in Fig. 5(a)), one identifies the band of interest (white rectangle) along with the fragments of the next neighbors (red shapes). After upsampling by a factor of 3, the signal is now available with 4 SPS, so that the Rx spectrum is repeated 2 more times (second from right in Fig. 5(a)). Interpolation with a sinc-function (blue dashed box) as described by the preceding paragraph would only remove the periodic repetitions (frequency response $\text{rect}(f)$: blue dashed rectangle) but not the fragments of the neighboring channels. These would lead to crosstalk during the subsequent downsampling process.

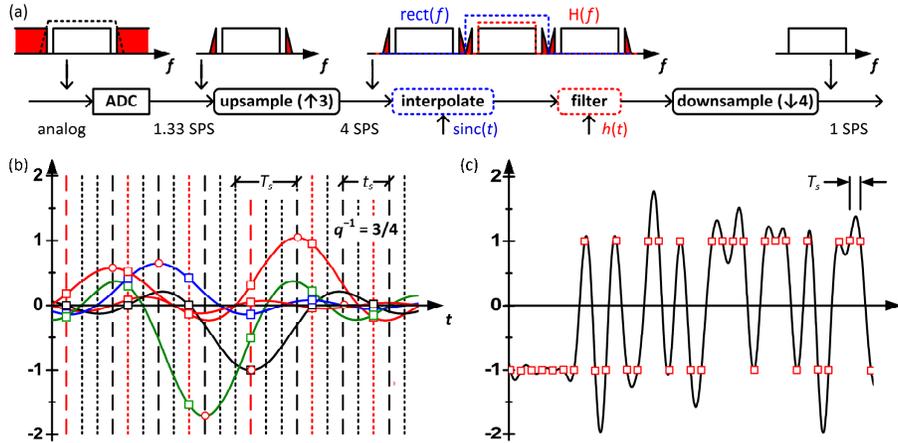


Fig. 5. Rx filter if neighboring channels are not fully removed by analog filters. (a) An ADC samples the analog signal with 1.33 samples per symbol (SPS). The spectrum shows fragments of the neighbors (red). Upsampling results in periodic repetitions. The interpolation filter ($\text{rect}(f)$, blue dotted line) removes the periodic repetitions, but not the neighboring fragments (red). To avoid crosstalk during downsampling, these fragments have to be removed by an additional filter ($H(f)$, red dotted line). Both filters can be combined and replaced by $\text{rect}(f)H(f) = H(f)$. (b) Combined Rx resampling and filtering by using the same technique as described by Fig. 4(c). (c) Waveform after combined resampling and filtering results in the same signal as in Fig. 4(d).

In order to remove these fragments, an additional filter (frequency response $H(f)$: red dashed rectangle) with IR $h(t)$ must be employed (red dashed box). Fortunately, the two filters can be combined to a single one having a frequency response $\text{rect}(f)H(f) = H(f)$, since $\text{rect}(f)$ is wider than $H(f)$. So, instead of using an interpolation as shown in Fig. 4(c), we can directly implement the filter $H(f)$ with its impulse-responses $h(t)$ according to Fig. 5(b). It should be noted that unlike in Fig. 4(c), the zeros of the weighed IRs do not coincide with the pulse centers. Naturally, the filter coefficients for the procedure Fig. 4 differ from the ones for $h(t)$ in Fig. 5, otherwise, there is no structural difference and the Rx can be as efficiently implemented as described in Fig. 4.

If no neighboring channels are present, both methods shown in Fig. 4(c) and Fig. 5(b) lead to the same result shown in Fig. 4(d) and Fig. 5(c). Furthermore, the filter $H(f)$ does not need to be rectangular and can have a square root raised-cosine shape as well.

We conclude that a significant amount of processing effort in Tx and Rx can be saved when filtering data with a rational q directly, instead of performing a conventional two-stage process (upsampling, interpolation, and downsampling) for rational oversampling factors. The technique is not limited to sinc-shaped Nyquist signaling, but can be equally well used to efficiently realize processing of signals with different pulse shapes (e.g. with a square root raised-cosine spectrum).

4. Parallel FIR filter design

To enable multi-gigabit processing in real-time, either ASICs or FPGAs are commonly used. While ASIC development is time consuming and costly, FPGAs are the best solution for prototyping. Employing FPGAs, multi-gigabit data streams can be processed at much reduced clock rates through parallelization. In this section we discuss a parallel FIR filter design suitable for FPGA and ASIC implementations. The design is made suitable for an adaptable rational oversampling factor q . This allows changing symbol rate and signal bandwidth without touching the sampling frequency.

Traditionally, FIR filters are described serially by a sequence of delay elements and $R + 1$ taps in-between. Signal samples $x[m]$ are weighed by the filter coefficients $h[r]$ and summed to form the output $y[m]$ [3] according to the discrete convolution

$$y[m] = \sum_{r=-R/2}^{+R/2} x[m-r]h[r], \quad m, r \in Z. \quad (2)$$

For a parallel realization of a filter described by Eq. (2) it should be first noted that each sample $x[m]$ at the filter input generates a weighed IR $x[m] h[r]$ represented by $R + 1$ coefficients $h[r]$ at the filter output, where m, r represent discrete points in time. For a multiple of input samples $x[m]$, the output $y[m]$ follows from a linear superposition of all properly delayed IRs. Due to this linearity, we first generate all IRs $x[m] h[r]$ individually as depicted in Fig. 6. If only a limited number of different input values $x[m]$ is possible (as it is true at the Tx for M -ary QAM signals), then the required multiplications can be pre-computed and the results stored in look-up tables (LUT) [3].

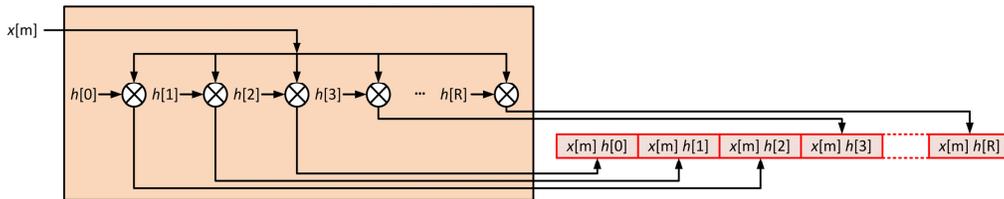


Fig. 6. Impulse response generator. A copy of the input sample is multiplied by each of the filter coefficients $h[r]$ representing the sampled IR of the filter with order R and $R + 1$ filter coefficients. The multiplication with $h[r]$ leads to a finite number of products $x[m] h[r]$ that can be stored in look-up tables (LUT) thereby avoiding resource-hungry multiplications.

With N copies of the IR generator of Fig. 6 we process N samples in parallel and hence reduce the clock speed of the DSP by a factor N . Processing of $N = 4$ samples per clock cycle (cycles are separated by dashed lines) is shown in Fig. 7. The clock waveform (purple) is depicted vertically on the left-most side of Fig. 7. Processing is performed at the rising edges of the clock. In each clock cycle $N = 4$ input samples $x[m]$ are fed to the IR generators of Fig. 6 leading to a block of $N = 4$ sampled and weighed IRs $x[m] h[r]$ (as depicted in the middle of Fig. 7). In this example, the IR of the filter is represented by $R + 1 = 8$ filter coefficients or filter taps (filter order $R = 7$). In order to prepare the computation of the filter output $y[m]$, the IRs are arranged in a two-dimensional (2D) array (right). After flipping the weighed IRs upside down, each IR within a row (elements $[r]$) is rotated counter-clockwise and mapped to a column (elements $[m]$) of the 2D array such that adjacent responses $x[m] h[r]$ are shifted by one sample with respect to each other. The output $y[m]$ is then obtained by summing all samples that are located in the same row of the 2D array.

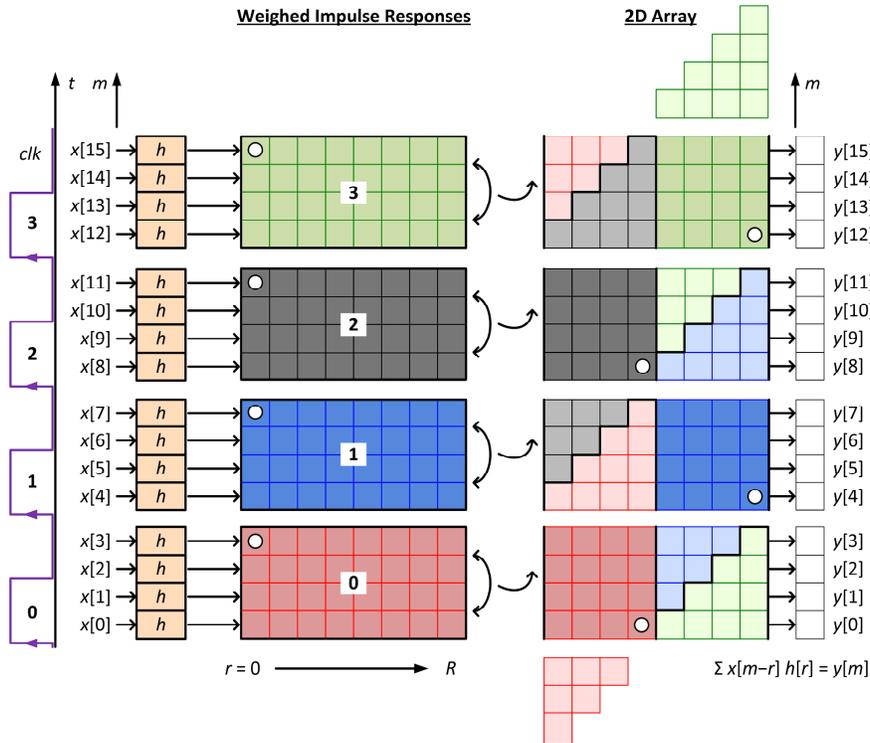


Fig. 7. Parallel FIR filter design where $N = 4$ input samples $x[m]$ are processed within each clock cycle (clk , left vertical time axis t) to produce four output samples $y[m]$ with the summation $y[m] = \sum x[m-r] h[r]$, Eq. (2). Each input sample is multiplied with the filter impulse response (IR) $h[r]$ using the IR generators described in Fig. 6. In this example, the filter response is represented by $R + 1 = 8$ filter coefficients. In order to prepare the computation of the filter output $y[m]$, the pulses are arranged within a two-dimensional (2D) array (right, flipped upside down and rotated counter-clockwise by 90°) with each IR at time m being delayed by one sample with respect to the previous IR at $m - 1$. The output $y[m]$ is then obtained by summing all samples that are located in the same row of the 2D array.

From Fig. 7 it can be seen that some samples of the weighed IRs spread into adjacent processing cycles. Buffer areas in the 2D array (lighter colors) are used to correctly account for these samples. To compute the $N = 4$ output samples of clock cycle 2 (black), input samples from the blue (cycle 1), black (cycle 2) and green (cycle 3) colored IRs need to be available. Therefore output values $y[8] \dots y[11]$ have to be computed within the following clock-cycle 3 leading to a latency of one cycle. In Fig. 7, it can be further seen that some samples of the red (cycle 0) and the green IRs (cycle 3) extend beyond the boundaries of the 2D array. These samples fit to the opposite sides of the array (rows $m = 13 \dots 15$ for the red samples and rows $m = 0 \dots 3$ for the green samples). With each additional clock cycle (4, 5, ...) the samples in the physical storage areas 0, 1, 2, 3 are replaced by the following scheme: cycle 4 \rightarrow area 0, cycle 5 \rightarrow area 1, cycle 6 \rightarrow area 2, cycle 7 \rightarrow area 3, cycle 8 \rightarrow area 0, ... Four storage areas were chosen in Fig. 7 for simplifying the explanation. However, since only samples from three cycles are required for computing $y[m]$, the 2D array needs containing only three physical storage areas 0, 1, 2 with $R + 1 = 8$ columns and 12 rows.

Besides processing multi-gigabit data streams in real-time, the described parallel FIR filter design allows changing the symbol rate and the bandwidth by varying the oversampling factor q during runtime. To this end, only the filter coefficients $h[r]$ in the IR generators of Fig. 6 (brownish boxes h in Fig. 7) need to be changed. The granularity with which q can be changed scales with the number N of samples being processed in parallel. This is similar to

the procedure for OFDM, where q is changed by nulling more or less of the outer SCs, and the granularity therefore scales with the number of SCs. To illustrate the flexible adaption of symbol rate and bandwidth through a change of q , we depict a single clock cycle 0 and its associated storage area 0 from Fig. 7 for $N = 4$ and $q = 1, 2, 4$, and $4/3$, see Fig. 8. The IR generators h are now subscripted with q for $q = 1, 2, 4$, and with A, B, C for $q = 4/3$. The situation in Fig. 7 with an oversampling factor $q = 1$ is reproduced in Fig. 8(a). For an oversampling factor $q = 2$, Fig. 8(b), the filter coefficients $h[r]$ of every other IR generator (Fig. 6) are set to zero. As a consequence, only every second input sample $x[m]$ transmits data, so that the data rate and the bandwidth of the transmitted signals is reduced by a factor of $q = 2$. In analogy to $q = 2$, an oversampling factor $q = 4$ is obtained by setting the filter coefficients $h[r]$ in three out of the four IR generators to zero, see Fig. 8(c). Finally, a rational oversampling factor $q = 4/3$ can be realized, see Fig. 8(d). This time the coefficients $h[r]$ in the IR generators are adjusted according to Fig. 4(a), where one distinguishes between three identical but differently sampled impulse responses (black, red, and blue).

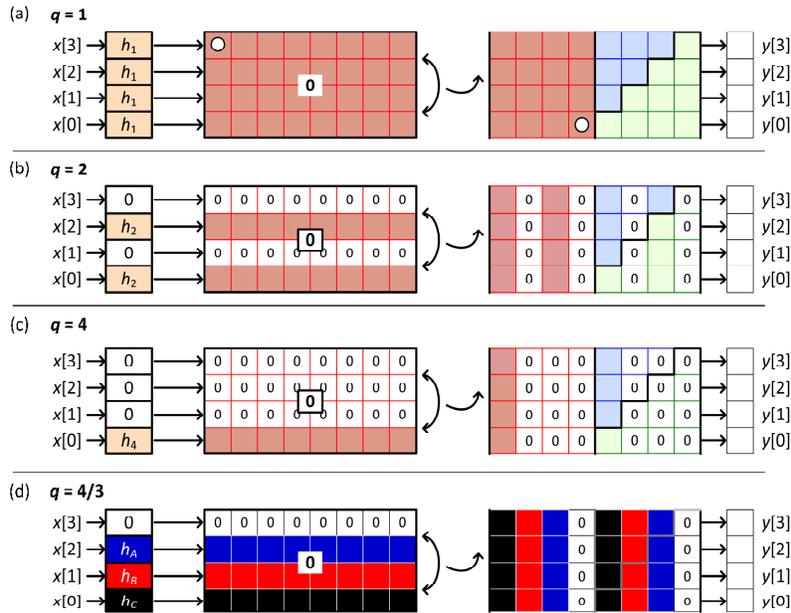


Fig. 8. Flexible adaptation of the symbol rate and bandwidth through changing the oversampling factor q while keeping the sampling rate f_s constant. For clarity, only clock cycle 0 according to Fig. 7 is shown. (a) Clock cycle 0 of Fig. 7 is reproduced, $q = 1$ (b) For an oversampling factor $q = 2$ every other IR generator must produce zeros as impulse response, thereby reducing the data rate by a factor of 2. (c) To generate signals with $q = 4$, three of the 4 pulses processed in parallel are set to zero. (d) A rational oversampling factor such as $q = 4/3$ can be realized by adjusting the IR generator coefficients $h_{A, B, C}$ according to Fig. 4(a).

The exchange of filter coefficients can be realized in a single clock cycle. It should be noted, however, that a change of q affects the duration $T = (R/q) T_s$ of the impulse response which is represented by a fixed number of $R + 1$ coefficients. For q close to 1, any infinitely extended impulse response is much better approximated than for a q close to N . This affects the steepness of the signal band slopes and thus the potential SE in a way similar to OFDM when nulling outer SCs to effectively increase q .

In order to prove the adaptability of our signal processing with respect to symbol rate and signal bandwidth we use a filter with order $R = 64$ and simulate the VHDL-description (very high speed integrated circuit hardware description language) on the register transfer level (RTL), a design abstraction which models the signal flow of synchronous digital circuits. While the filter coefficients are varied to cover the three practical scenarios indicated in Fig.

8(b)–8(d), the clock rate remains constant. From the generated signals we derive the ensemble-averaged power spectra using a fast Fourier transform (FFT). Figure 9 shows the averaged power spectra as a function of frequency normalized with respect to the sampling frequency f_s . As expected, the spectrum for $q = 4$ is confined between $-0.125 f_s$ and $0.125 f_s$, i.e., the bandwidth is reduced to $f_s / 4$, see Fig. 9(a). Changing the filter coefficients for an oversampling factor $q = 2$ increases the symbol rate and the bandwidth to $f_s / 2$, see Fig. 9(b). Finally, we adapt the filter coefficients for an oversampling of $q = 4 / 3$ and obtain a spectrum confined between $-0.375 f_s$ and $0.375 f_s$ and thus a signal bandwidth of $f_s / (4 / 3)$, Fig. 9(c).

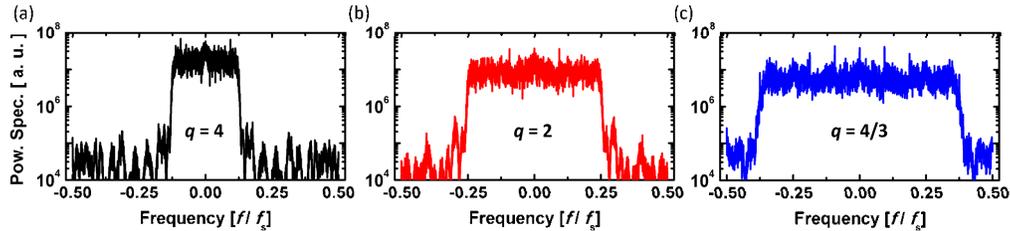


Fig. 9. Ensemble-averaged power spectra obtained from VHDL simulations as a function of frequency normalized to the sampling rate f_s . All Nyquist signals are generated with the same FIR filter structure of order $R = 64$. (a) Spectrum for a signal oversampling of $q = 4$. The signal bandwidth is $f_s / 4$. (b) Spectrum for an oversampling $q = 2$. The signal bandwidth is $f_s / 2$. (c) Spectrum for an oversampling $q = 4 / 3$. The signal bandwidth is $f_s / (4 / 3)$.

Resource requirements: The previously described real-time parallel processing has been implemented in VHDL, synthesized, and then evaluated in terms of resources. We implemented FIR filters of different orders R and with an oversampling factor $q = 4 / 3$. The corresponding impulse response duration T for a filter of order R is $T = (R / q) T_s$. The higher the order R , the better a sinc-impulse response is approximated [3]. All evaluated filter designs are optimized with respect to minimum resource requirements and maximum speed, i.e., no flexible adaptation of q is supported at this point. Since we used the filters for an M -ary QAM Tx, we realized the IR generators of Fig. 6 with LUTs for avoiding multiplications. For the final FPGA design a number of $N = 128$ samples are processed in parallel. Table 1 shows the resource utilization of the FPGA for a Nyquist pulse-shaper of order R that can flexibly switch between QPSK and 16QAM. We depict the most significant resources, i.e., the number of slice registers and slice LUTs. To give a quantitative estimate of what can be achieved with state-of-the-art FPGAs, we relate the resources to one of the biggest available Virtex 7 FPGAs (XC7VX980T). In this work, however, we were restricted by the number of slice registers and slice LUTs provided by two Virtex 5 FPGAs (XC5VFX200T). Therefore we used a filter of order $R = 32$ for the experiments discussed later on, since it can be handled by the Virtex 5 FPGA while still meeting the timing constraints with on-chip clock frequencies of up to $f_s / N = 28\text{GHz} / 128 = 218.75$ MHz.

We now compare the resource requirements of processing with the standard two-fold oversampling [3, 10] q_2 to processing using (4/3)-fold oversampling $q_{4/3}$. For a filter order $R = 128$ and a degree of parallelization $N = 128$ we find that the number of slice registers increases by a factor of 2.8 and the number of slice LUTs increases by a factor of 2.7. However, the number of symbols processed within each clock cycle also increases by a factor of $q_2 / q_{4/3} = 3 / 2$, and so does the duration T of the impulse response. With these numbers we calculate the amount of slice registers and slice LUTs required to process a single symbol for a fixed impulse response duration T . We find that a 1.5-fold (50%) increase in speed comes at the moderate price of a 14% (slice registers) and 17% (slice LUTs) increase of resources.

Table 1. FPGA Resource Utilization for FIR Filters of Different Orders R and Impulse Response Duration T

Resource	$R = 16$ ($T = 12 T_s$)	$R = 32$ ($T = 24 T_s$)	$R = 64$ ($T = 48 T_s$)	$R = 128$ ($T = 96 T_s$)
Slice registers (% of XC7 / $2 \times$ XC5)	29,300 (2% / 12%)	60,120 (5% / 24%)	109,940 (9% / 45%)	210,320 (17% / 86%)
Slice LUT (% of XC7 / $2 \times$ XC5)	25,180 (4% / 10%)	50,360 (8% / 20%)	93,120 (15% / 38%)	175,080 (29% / 71%)

5. Experimental setup

The real-time Nyquist pulse Tx, Fig. 10, comprises two synchronized Xilinx XC5VFX200T FPGAs, two Micram 6 bit DACs, optional 12.3 GHz image rejection low-pass filters, and an optical I/Q-modulator. Payload data is emulated by a pseudo random binary sequence (PRBS) with a periodicity of $2^{15}-1$. The generated Nyquist sinc-pulses are encoded on an external cavity laser (ECL) operated at a center wavelength of 1550 nm.

For experiments with transmission (switch position 1), the link is formed by an erbium doped fiber amplifier (EDFA 1) together with a polarization emulation setup where the signal is split, delayed by 5.4 ns with respect to each other, and combined in orthogonal polarizations. Measurements to evaluate the performance of the dynamic computational precision are performed with a transmission link of up to four 75 km spans of SSMF (attenuation: 0.19 dB / km, dispersion: 16.8 ps / nm / km) with in-line EDFA amplification. For the investigation of PDM-64QAM oversampled by $q = 4 / 3$, the transmission link comprises 100 km ULAF (attenuation: 0.18 dB / km, dispersion: 20.2 ps / nm / km).

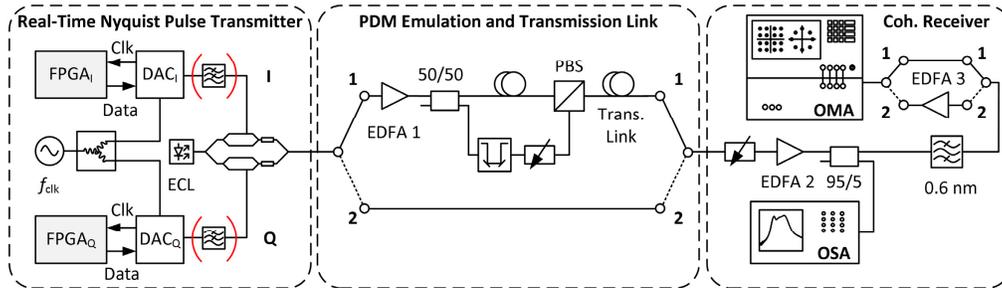


Fig. 10. Experimental setup for real-time Nyquist sinc-pulse shaping. The Tx comprises two synchronized FPGAs, two DACs, optional image rejection filters, and an optical I/Q-modulator. Data are encoded on an ECL at 1550 nm. For transmission experiments (switch position 1): EDFA 1 and PDM emulation followed by either up to four 75 km spans of SSMF and EDFAs (for the evaluation of the dynamic computational precision) or by 100 km ULAF (for evaluation of $q = 4 / 3$ oversampled PDM-64QAM transmission). Signals are recovered with a coherent receiver (EDFA 2 and Agilent OMA). For QPSK and 16QAM measurements with oversampling factor $q = 4 / 3$ only, switch position 2 bypasses the link. The OSNR is adjusted by varying the input power to EDFA 2. In this case, we keep the OMA at optimum input power with a gain-controlled EDFA 3.

The input power to the Rx EDFA 2 is adjusted for sweeping the optical signal-to-noise ratio (OSNR) (switch position 2), measured with an optical spectrum analyzer (OSA) (reference bandwidth: 0.1 nm). A 0.6 nm filter removes out-of-band EDFA noise. For operating the Agilent optical modulation analyzer (OMA) optimally, EDFA 3 is employed (switch position 2). Finally, the OMA coherently receives the signals using a second, inbuilt ECL and real-time oscilloscopes. Further processing including error vector magnitude (EVM) and bit error ratio (BER) computation [16] is performed offline.

6. Experimental results

In the following we present the results obtained with the setup of Fig. 10. First, we investigate the performance of dynamically adjusted precision for PDM-16QAM and PDM-64QAM signals transmitted over up to 300 km of SSMF. Second, Nyquist signals are generated with an oversampling factor $q = 4 / 3$. We vary the sampling rate of the DACs resulting in QPSK and 16QAM Nyquist signals with symbol rates of 15 GBd, 18.75 GBd, and 21 GBd. We further demonstrate PDM-64QAM at 21 GBd and transmit the signal over 100 km ULAF. For all measurements we record the BER counting at least 100 errors per measurement and/or the EVM derived from >50,000 received symbols.

6.1 Dynamic computational precision

Finite duration impulse response filters with conventional, “standard” DSP (fixed effective word length) and with dynamically adjusted DSP word length (with variants \div and \times) are implemented on the register transfer level (RTL) using VHDL. All filters in this section are of order $R = 64$ and process two-fold oversampled signals ($q = 2$). The overall chip (FPGA) utilization for the filters with different DSP precision is depicted in Table 2.

Table 2. FPGA Resource Utilization for FIR Filters with and without Dynamically Adjusted DSP Word Length

Resource	Fixed word length (WL)	Dynamically adjusted WL \div -variant	Dynamically adjusted WL \times -variant
Slice registers (% of XCV5FX200T)	39,040 (32%)	39,040 (32%)	36,600 (30%)
Slice LUT (% of XCV5FX200T)	31,250 (25%)	32,500 (26%)	30,000 (24%)

Subsequently, the filters are employed in our real-time Nyquist Tx. Measured formats include 14 GBd PDM-16QAM and 12.5 GBd PDM-64QAM. Nyquist pulses are transmitted over up to four spans of 75 km SSMF. A measured (black) and VHDL-simulated (white) spectrum of a 16QAM signal is depicted in Fig. 11(a). The signal bandwidth is close to 14 GHz for a symbol rate of 14 GBd. The observed noise floor is due to quantization noise of the 6 bit Tx DACs and the 8 bit Rx ADC (OMA oscilloscope). We further measure the BER for PDM-16QAM and PDM-64QAM signals, for all filter implementations and for different fiber spans, Fig. 11(b). It can be concluded from Fig. 11 and Table 2 that signals generated with dynamically adjusted word length from 6 bit to 11 bit give superior signal quality without increasing the processing complexity. As expected, the (\times)-filter shows best performance requiring similar resources on the FPGAs as the standard fixed precision DSP.

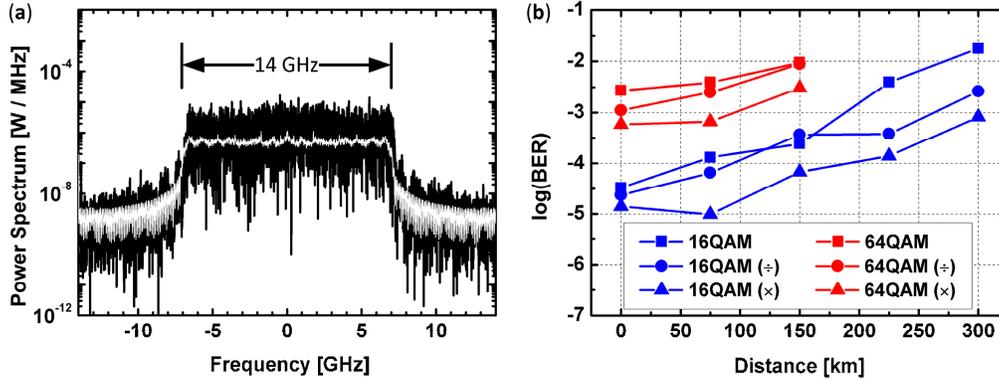


Fig. 11. Experimental results for Nyquist shaped PDM-16QAM (14 GBd) and PDM-64QAM (12.5 GBd) transmitted over various spans of SSMF. (a) Measured (black) and simulated (white) spectra for 16QAM signals. The spectral noise floor is due to quantization noise, and can be removed by analog low-pass filters. (b) Measured BER of both modulation formats and fiber spans for standard and dynamically adjusted DSP word lengths as described in Fig. 3(c). As expected, the (\times)-design shows best performance.

6.2 Rational oversampling

We measure the spectra, the BER, and we estimate the BER from the measured EVM [16] for Nyquist sinc-pulses generated with a real-time pulse-shaper ($R = 32$) and with an oversampling factor $q = 4 / 3$ for various OSNR, Fig. 12. Results obtained with (without) image rejection filters are colored black (red). The measured signal spectra in Fig. 12(a) show an out-of-band suppression of 30 dB and a bandwidth close to 21 GHz for 21 GBd signals. The image spectra Fig. 12(a) (red) are effectively removed by the filters (black). Figure 12(b)-12(d) depicts the measured BER (squares) and BER as calculated from measured EVM (lines). For symbol rates beyond 15 GBd the available image rejection filters introduce a penalty due to the non-optimum cut-off frequency of 12.3 GHz. Due their suboptimal frequency response [17] for Nyquist signals with these symbol rates, measurements in Fig. 12(c) and 12(d) are taken solely without filters. Recently, it has been shown that carefully designed low-pass filters (LPF) allow for oversampling factors as low as $q = 1.15$ [18].

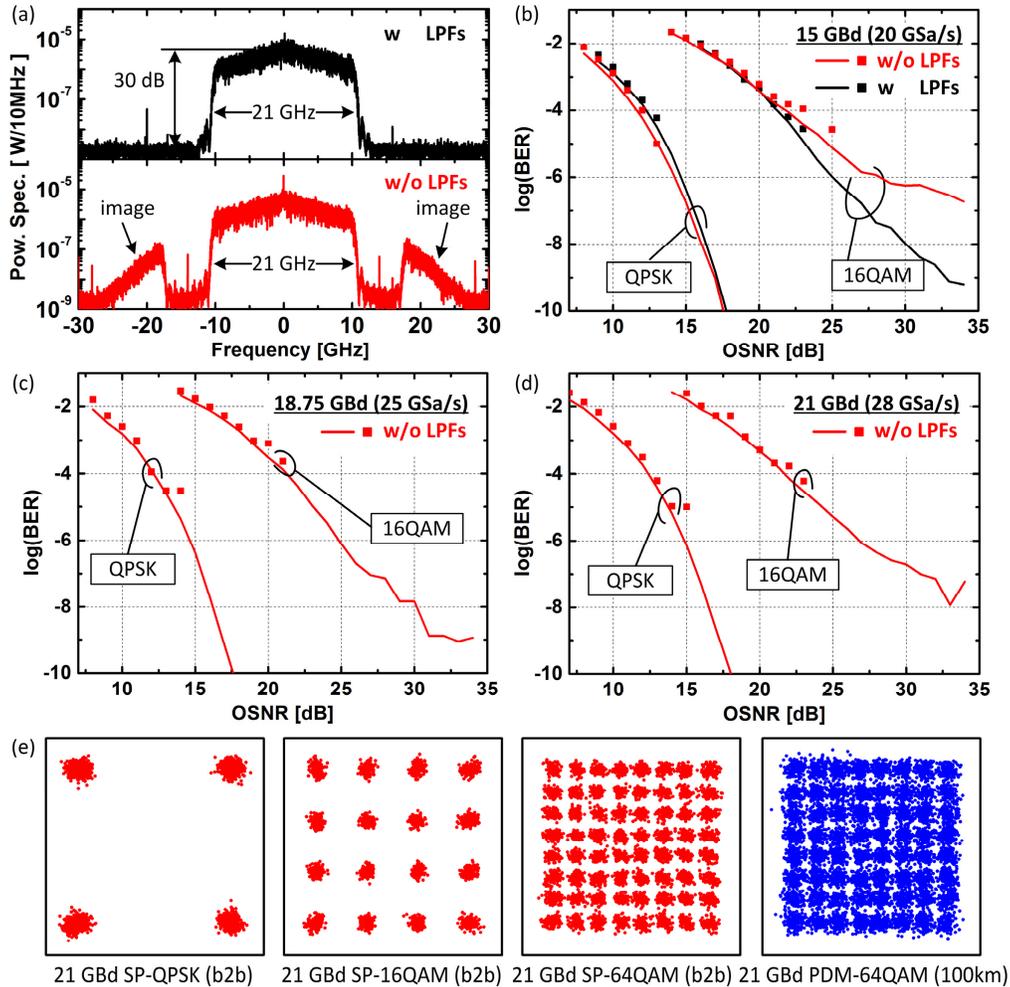


Fig. 12. Experimental results for real-time Nyquist pulse shaping with filter order $R = 32$. (a) 21 Gb/s 16QAM spectra measured with low pass filters (LPF) used for image rejection (black) and without (red). (b) – (d) BER (squares) and BER estimated from measured EVM (lines) as a function of OSNR for QPSK and 16QAM with (black) and without (red) image rejection filters. (e) Constellation diagrams obtained back-to-back for a single polarization (SP, red) and for PDM-64QAM after transmission over 100 km (blue).

Figure 12(e) shows constellation diagrams for 21 Gb/s single polarization (SP) signals (red) at highest possible OSNR. For SP-64QAM (red) and in a back-to-back configuration, a $BER = 1.84 \times 10^{-3}$ is achieved. After 100 km transmission of PDM-64QAM (blue) at a bit rate of 252 Gbit/s the resulting raw BER of 1.43×10^{-2} can still be improved by state-of-the-art forward error correction (FEC) [18].

7. Conclusions

Flexible DSP for real-time Nyquist signaling realized with state-of-the-art FPGAs has been investigated. In particular, we demonstrated how a dynamic computational precision enhances the performance without increasing the FPGA resource requirements. We further demonstrated real-time Nyquist signaling with a rational oversampling factor $q = 4 / 3$ and showed that varying the symbol rate and bandwidth can be achieved by flexibly adjusting this oversampling factor. Finally, we confirmed all results with experiments. To this end, QPSK,

16QAM and 64QAM Nyquist signals were evaluated. Signals with a maximum data rate of 252 Gbit/s were transmitted over 100 km ULAF with a BER below the soft-decision FEC limit.

Acknowledgments

This work was supported by the projects FOX-C, ACCORDANCE, and CONDOR, by Micram Microelectronic GmbH, OFS, Piano + OTONES as well as by the Xilinx University Program (XUP), and by the Agilent University Relations Program. We further acknowledge financial support by Karlsruhe School of Optics & Photonics (KSOP), by the Deutsche Forschungsgemeinschaft, and the Open Access Publishing Fund of the Karlsruhe Institute of Technology (KIT).