Real-Time DSP-Enabled Digital Subcarrier Cross-Connect Based on Resampling Filters

Tong Xu, Andrea Fumagalli, and Rongqing Hui

***Abstract*—A real-time digital signal processing (DSP)-enabled digital subcarrier cross-connect (DSXC) with resampling filters for channel frequency translation is demonstrated in this paper. This circuit-based DSXC supports flexible and fine data-rate subcarrier channel granularities, offering a low-latency data plane, transparency to modulation formats, and the capability of compensating for transmission impairments in the digital domain. The presented 8 × 8 DSXC demonstrator makes use of a Virtex-7 field programmable gate array (FPGA) platform, which supports any-to-any switching of eight subcarrier channels with mixed modulation formats and data rates. Digital resampling filters enable frequency translation of subcarrier channels at the cross-connect with much reduced processing requirements for the FPGA resources in comparison to the traditional technique based on in-phase/quadrature mixing and filtering.**

***Index Terms*—DSXC; Frequency translation; Real-time DSP; Resampling filters.**

I. Introduction

Due to the ever-increasing data traffic in today's optical transport networks, efficient utilization of the optical spectrum is essential. A technique to improve spectral efficiency (SE) is based on an elastic optical network (EON). EONs advocate the use of a flexible spectrum grid in place of the fixed International Telecommunication Union (ITU) grid of conventional optical wavelength division multiplexing (WDM) networks [1]. Both bandwidth and channel allocation are flexible in EONs and can be chosen to best accommodate the modulation formats of choice, transmission distance, system capacity, and number of required channels [2–4]. However, due to the coarse bandwidth granularity imposed by optical filtering, an optical domain EON alone may not be sufficient to achieve the fine bandwidth allocation that may be required in some access and metro-area network deployments. For this and other reasons, conventional solutions combine the use of optical circuits with electronic packet-switching technologies [Ethernet, IP, passive optical network (PON)], i.e., multiple connections are multiplexed together by interleaving their data packets in time, thus filling up the relatively large bandwidth of the optical circuit.

Subcarrier multiplexing (SCM) [5,6] represents a promising alternative to electronic packet switching in some access and metro deployments. Subcarrier circuits can be flexibly multiplexed and individually switched electronically, offering dedicated circuits to the application down to megahertz (MHz) of bandwidth. Through the use of a commercially available high-speed analog-to-digital converter (ADC), digital-to-analog converter (DAC), and digital signal processing (DSP) modules, SCM can nowadays be implemented digitally, a technique that is known as digital subcarrier multiplexing (DSCM) [7]. Accurate and flexible DSP algorithms can be applied to achieve DSCM of complex high-level modulation schemes and reduced requirements for the spectral guard band between subcarrier channels. Orthogonal frequency-division multiplexing (OFDM) and Nyquist frequency-division multiplexing (N-FDM) are well-known examples of DSCM with high spectral efficiency [8].

To individually switch and route digital subcarriers, the authors recently introduced the concept of digital subcarrier cross-connect (DSXC) [9,10]. Compared to optical cross-connect (OXC)-based technologies (either fixed or flexible spectrum), DSXC provides a much finer switching bandwidth granularity [9,10]. Compared to electronic packet switching technologies, DSXC has the unique advantage of concurrently supporting and enabling the multiplexing of a variety of radio-over-fiber (RoF) transmission technologies, including the switching of DSP-enabled analog signals.

With the beginning of the 5G and Cloud Radio Access Network (C-RAN) era, the design of mobile network fronthaul—the network segment connecting the radio unit (RU) to the baseband unit (BU)—has drawn increasing attention [11–13]. RoF systems based on analog transmission offer improved SE and can greatly reduce the number of physical links in the network fronthaul [14]. An efficient 5G mobile fronthaul based on real-time DSP-enabled channel aggregation/de-aggregation has been recently demonstrated [15,16]. The introduction of DSXC with real-time dynamic bandwidth allocation and transparency to modulation formats can further and greatly enhance the RoF capabilities and functionalities of the mobile fronthaul.

Manuscript received August 8, 2018; revised September 13, 2018; accepted September 18, 2018; published October 12, 2018 (Doc. ID 341846).

T. Xu and R. Hui (e-mail: rhui@ku.edu) are with the Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, Kansas 66044, USA.

A. Fumagalli is with the Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, Richardson, Texas 75083-0688, USA.

https://doi.org/10.1364/JOCN.10.000937

In recent years, the time-sensitive network (TSN) has become an important topic of investigation, not only for C-RAN applications, but also for other emerging applications such as cloud-based robot control, tele-surgery, and network gaming [17]. These applications demand a tight network round-trip delay and jitter requirements, and TSN standards have been identified to ensure that the packet-switching network (e.g., Ethernet) meets said stringent requirements. By its own design, DSXC is a technology that is well suited to provide minimal round-trip delay and virtually zero jitter. In comparison to a packet-switched network solution, which normally requires packet store-and-forward processing along with packet buffering at the network switches, the DSXC circuit-based solution enables incoming data bits to be continuously forwarded to the DSCN node output port. For example, current data center networks consisting of electronic packet switches may experience relatively large network latency due to packet store-and-forward processing [18]. This latency, caused by packet queuing and processing time, is often dynamic and unpredictable.

Both OFDM and N-FDM have been efficiently achieved through real-time DSP implemented in field-programmable gate arrays (FPGAs) [19,20]. DSP-enabled real-time reconfigurable optical add/drop multiplexing (ROADM) technologies have also been demonstrated using FPGA platforms [21,22]. However, real-time DSXC experimental results that demonstrate cross-connect and switching capabilities of multiple subcarrier channels of arbitrary waveforms' flexible bandwidth have not yet been reported.

In this paper, the authors describe the first implementation of a DSXC node, realizing one of its basic functionalities (switching individual subcarriers in frequency) using digital resampling filters and experimentally assessing the DSXC node (deterministic) latency to be less than 1 μs. This latency is mainly determined by the order of finite impulse response (FIR) filters and the clock period of the digital circuit. Compared to commodity packet switches, DSXC may therefore provide a simple and cost-effective switching solution that achieves zero jitter, even in the presence of high link utilization.

II. Digital Subcarrier Cross-Connect Concept

Figure 1 shows a basic DSXC block diagram. The input signal to the DSXC switch fabric comprises n wavelength channels, and each wavelength carries m radio frequency (RF) subcarrier channels. The RF subcarriers can support different modulation formats and make use of different spectral bandwidths. Each wavelength channel is detected by an optical receiver performing optical-to-electrical (O/E) conversion, which is followed by an ADC digitizing the analog waveform delivered by the optical carrier. Then the digitized waveform is sent to a DSP block for subcarrier demultiplexing. The de-multiplexed subcarriers from all wavelength channels are sent to a cross-bar circuit switch, in which any input subcarrier can be routed to any output port. Subcarrier channel local add/drop can also be performed in this cross-bar switch unit. The subcarriers switched through the cross-connect are then re-grouped and multiplexed by n DSP units corresponding to n output wavelength channels. Each digitally multiplexed composite signal forms a wavelength channel that is converted from electrical to optical (E/O) through a DAC and an optical transmitter. With this basic DSXC architecture, any subcarrier channel of any input optical carrier can be routed to any subcarrier of any output wavelength channel.

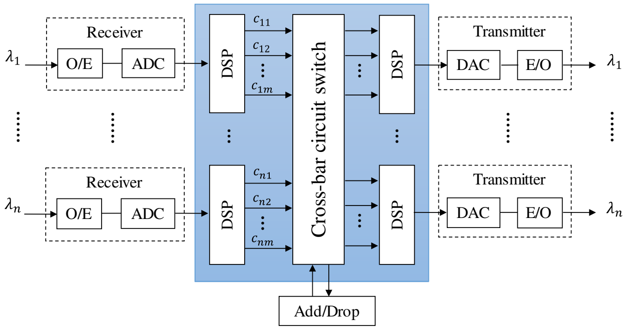


Fig. 1. Block diagram of digital subcarrier cross-connect (DSXC).

For demonstration purposes, the DSP units and the cross-bar circuit switch can be implemented using a single FPGA module that provides real-time processing. The cost of this implementation choice is proportional to the amount of FPGA resources required to implement the DSXC. FPGA resources mainly consist of memory resources and DSP slices. Memory resources include look-up tables (LUT), LUT random access memory (LUTRAM), flip-flop (FF), and block RAM (BRAM). DSP slices are used to carry out digital multiplications, which are usually the most costly operation in a real-time DSP hardware platform. For convenience, in this paper we use the term *DSP cost* to represent the number of required DSP slices.

Since subcarrier channels in each wavelength are multiplexed in the frequency domain, frequency translation (also known as spectral translation) is a critical operation in DSXC. A straightforward and conventional frequency-translation technique is based on signal mixing and filtering operations whereby multiple local oscillators (LOs), mixers, and low-pass filters are combined to achieve the intended goal. Digital filtering, which is the convolution between the input data sequence and the filter coefficients, is achieved through FIR filters that involve a large number of multiplications and represent the major DSP cost. Because the number of FIR filters increases linearly with the number of subcarriers in the cross-connect switch, the DSP cost for a DSXC based on this frequency-translation technique also increases accordingly and may become a major limiting factor.

Since DSXC is performed in the digital domain, we can utilize some inherent properties of digital sequence and DSP algorithms to achieve frequency translation at a reduced DSP cost. More specifically, interpolation and decimation are techniques that have been widely used in digital systems to change the sampling rate of a signal [23]. By applying some modification as described in Section III, interpolation and decimation techniques can be used to perform frequency translation with significantly less DSP cost in comparison to the frequency translation obtained through conventional in-phase/quadrature (I/Q) mixing and filtering. For the reader's convenience, we first briefly describe frequency translation through I/Q mixing and filtering. Then, we describe frequency translation through resampling filters.

III. Frequency-Translation Techniques

In this section, we briefly introduce two techniques of frequency translation: (1) I/Q mixing and filtering; (2) resampling filters.

A. *I/Q Mixing and Filtering*

Traditionally, frequency translation, which includes down-conversion and up-conversion, is achieved through I/Q mixing and filtering as shown in Fig. 2, in which LOs are implemented from direct digital synthesizers (DDS). In order to maintain phase synchronization between the LO and the RF subcarrier whose frequency needs to be translated, I/Q mixing is usually required. Figure 2 shows a standard two-step digital frequency-translation process that consists of both digital down-conversion (DDC) and digital up-conversion (DUC). A more detailed description of I/Q mixing and filtering is given in Appendix A.

As previously noted, when implementing this frequency translation in a FPGA platform, the major DSP cost comes from digital filters, which increases linearly with the number of subcarrier channels.

B. *Resampling Filters*

Alternatively, frequency translation may be achieved through resampling and filtering of each subcarrier channel. Similar to mixing and filtering, the resampling and filtering technique also consists of both DDC and DUC processes. As shown in Figs. 3(a) and 3(c), DDC is usually achieved by cascading a band-pass filter (BPF) with a down-sampling unit, while DUC can be achieved with an up-sampling unit followed by a BPF. For the sake of simplicity, we refer both down-sampling and up-sampling as resampling [23].

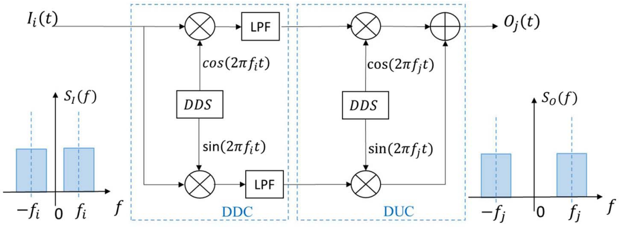


Fig. 2. Frequency translation through I/Q mixing and filtering.

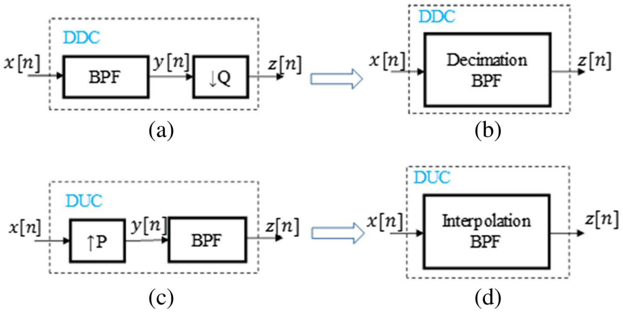


Fig. 3. (a) DDC through BPF and down-sampling, (b) DDC through interpolation BPF, (c) DUC through up-sampling and BPF, and (d) DUC through decimation BPF.

In Fig. 3(a), a BPF is used to select a specific subcarrier channel, and the data sequence after BPF is down-sampled by a factor of Q through a down-sampling unit.

Figure 4 shows an example of digital Fourier transform (DFT) spectra of the input data sequence $y[n]$ and the output data sequence $z[n]$ of the down-sampling unit with an input sampling frequency F_s and a down-sampling factor $Q = 4$. Through down-sampling, the frequency range is scaled down by a factor of 4 from $(-F_s/2, F_s/2)$ to $(-F_s/8, F_s/8)$. As shown in Fig. 4(a), the selected subcarrier channel originally located in frequency slot 2 (FS2) is automatically down-shifted to the frequency slot $(0, F_s/8)$. In general, if the selected subcarrier channel is originally located within an even frequency slot, such as FS2 and FS4 in Fig. 4(a), it will be spectrally flipped after down-sampling, while if it is located in an odd frequency slot, such as FS1 and FS3, its spectrum will not flip.

For the process of DUC shown in Fig. 3(c), a subcarrier channel at the lowest frequency slot needs to be translated to a higher frequency slot. Figure 5 shows an example of DFT spectra of the input data sequence $x[n]$ and the output data sequence $y[n]$ of the up-sampling unit with a sampling rate F_s and an up-sampling factor of $P = 4$. Through up-sampling, the frequency range is expanded 4 times from $(-F_s/2, F_s/2)$ to $(-2F_s, 2F_s)$. The up-sampled DFT spectrum in this expanded frequency range consists of multiple copies of the original spectrum, and each of them falls into a different frequency slot. By applying a band-pass filter on the up-sampled spectrum, a particular copy of the spectrum at the desired frequency slot can be selected, which is equivalent to a frequency translation. Again, similar to the down-sampling process, the frequency-translated spectra in even frequency slots, such as FS2 and FS4 shown in Fig. 5(b), are flipped in comparison to the original

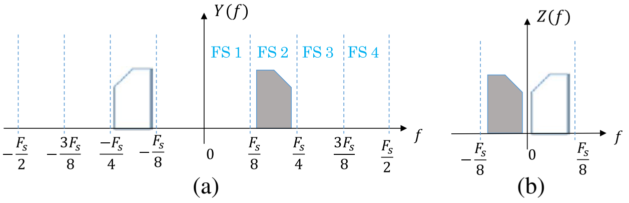


Fig. 4. (a) DFT of $y[n]$ and (b) DFT of $z[n]$ for down-sampling.

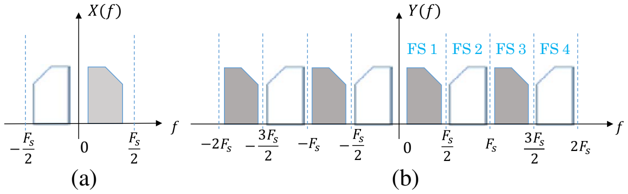


Fig. 5. (a) DFT of $x[n]$ and (b) DFT of $y[n]$ for up-sampling.

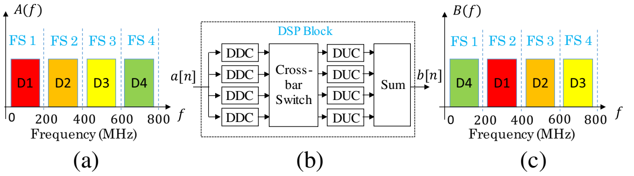


Fig. 6. (a) Spectrum of $a[n]$, (b) DSP block of DSXC, and (c) spectrum of $b[n]$.

spectrum in FS1. The flipped spectrum, although it contains the full information, is a frequency-conjugated version of the original signal, and thus another conjugate operation has to be performed when the baseband waveform needs to be recovered.

In comparison to I/Q mixing and filtering, the resampling and filtering technique shown in Figs. 3(a) and 3(c) does not need LOs and mixers, and there is no need for carrier phase synchronization. Since the actual bandwidth of each frequency slot is determined by the sampling frequency and the resampling factor, it can be flexible to accommodate different data rates carried by different subcarriers. Suppose the sampling frequency is f_s and the resampling factors are L_1, L_2, and L_3, then the bandwidths of the frequency slot after resampling are $f_s/(2L_1)$, $f_s/(2L_2)$, and $f_s/(2L_3)$, respectively. The channel data rate granularity of DSXC can be made fine enough to address network efficiency requirements through the change of the resampling factor.

However, the major drawback of both DDC and DUC shown in Figs. 3(a) and 3(c) is that the BPF still requires significant DSP resources of FPGA, similar to those based on I/Q mixing and filtering. A novel technique to solve this problem is to combine the resampling and BPF into a single resampling BPF as shown in Figs. 3(b) and 3(d). The resampling BPF is a general term that includes the decimation BPF for DDC and the interpolation BPF for DUC.

Resampling filters could be implemented as polyphase decimation or interpolation filters on FPGA hardware [24], which was proposed primarily for resampling of data sequences while avoiding spectral aliasing and rejecting spectral images. Although polyphase resampling filters have been previously used in wireless transceivers [25], they have not been used for DSXC switches, which require the capability of handling asynchronous subcarrier channels with non-equal bandwidth and independent modulation formats. While the required DSP resources linearly increase with the number of subcarrier channels for both I/Q mixing and filtering and resampling and filtering, the DSP resources required for a resampling BPF are independent of the number of subcarrier channels. This significantly reduces the DSP resource requirement for FPGA implementation. More detailed descriptions of the resampling BPF can be found in Appendix B.

IV. Resource Cost of Frequency Translation

In order to estimate the resource cost of different frequency-translation techniques, a 4 × 4 DSXC is designed in a Xilinx System Generator. In this design, the FPGA platform is based on Xilinx Virtex-7 690t [26].

Figure 6(a) shows an example of an input electrical signal spectrum that has four subcarrier channels each carrying a different data sequence ($D1 \sim D4$). For the sake of simplicity, in this example each subcarrier channel has the same bandwidth. Figure 6(b) shows the block diagram of DSP used for this 4 × 4 DSXC. In this DSP block, the composite digital sequence including all four subcarrier channels at the input is first made into four equal copies. Each of the four DDC blocks down-converts a channel from its subcarrier frequency to the baseband. The 4 × 4 cross-bar switch routes each down-converted baseband data sequence to a DUC block for frequency up-conversion. Channel add/drop is also possible at this stage before DUC. After up-conversion with each channel assigned a new subcarrier frequency, these subcarrier channels are combined at the output and sent to a DAC. The spectrum of the output electrical signal is illustrated in Fig. 6(c) with the frequencies of subcarriers switched in comparison to the input spectrum.

Two frequency-translation techniques, one based on a resampling BPF, and the other one based on I/Q mixing and filtering, are compared for this example. Both of them use 800 MHz total analog bandwidth that is equally divided into four frequency slots with 200 MHz bandwidth in each slot. 40 MHz is reserved as the guard band between adjacent subcarrier channels. 79th-order finite impulse response (FIR) filters are used for both techniques with 60 dB stopband attenuation. For the resampling-based frequency-translation technique, the resampling factor is 4, and band-pass filtering in each DDC block is accomplished by a decimation BPF. Similarly, each DUC block also performs band-pass filtering, which is implemented as an interpolation BPF. For the frequency-translation based on mixing with LO, each DDC is performed by I/Q mixing and filtering by two low-pass FIR filters for the I and the Q channels, and each DUC also uses a DDS and two mixers, as shown in Fig. 2. In this configuration, LOs are implemented through DDS by using LUT, and mixers are implemented as digital multipliers; they both cost FPGA resources in memories and DSP slices. Figure 7 shows the comparison of the FPGA resource cost to build this 4 × 4 DSXC based on the two different frequency-translation techniques. In order to achieve the same performance, the DSXC based on I/Q mixing and filtering has more than twice the DSP cost than the one based on resampling filters.

Here we used a Xilinx Virtex-7 690t FPGA chip as the DSP hardware platform, and the total number of available

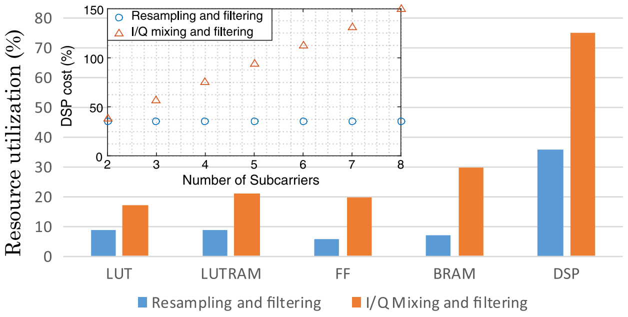


Fig. 7. FPGA resource cost of a 4 × 4 DSXC.

DSP slices on this chip is 3600. As indicated by Fig. 7, the bottleneck of the FPGA available resources in the design of DSXC is the DSP slices, so it is very important to minimize the cost of DSP slices in the design of the DSP algorithms. The usage of DSP slices is mainly consumed by FIR filters, and the design of FIR filters is a trade-off between the performance and resource cost. A higher-order FIR filter has smaller passband ripple, sharper cutting edges, and higher stopband attenuation, but has a higher resource cost. Passband ripple of an FIR filters causes frequency-dependent attenuation of the signal in the passband, which introduces signal waveform distortion.

For a traditional FIR filter to be implemented on a Virtex-7 FPGA, and supposing the length of its coefficients is S and the coefficients are symmetric, and if the degree of parallelism is R, then the DSP cost of this FIR filter is $R \times (S/2)$. Since a mixer is just a multiplier that supports parallel processing, it simply uses R DSP slices. For the I/Q mixing and filtering technique shown in Fig. 2, the frequency translation of each subcarrier channel needs two filters and four mixers, so that it requires $(S+4)R$ DSP slices. If the number of subcarrier channels is L, the total DSP cost of a DSXC based on I/Q mixing and filtering is

$$C_1 \approx (S+4)LR. \tag{1}$$

Since usually $S \gg 4$, the resources cost of this DSXC mainly comes from FIR filters. According to Eq. (1), the total DSP cost increases linearly with the number of subcarriers.

Resampling BPF will cost fewer resources compared to a traditional FIR filter of the same coefficients. In order to compare with the frequency translation based on I/Q mixing and filtering with the number of subcarrier channels of L, we assume the total available bandwidth is B, and the bandwidths of subcarrier channels are $B_1, B_2, \ldots, B_L$, so that $\sum_{i=1}^{L} B_i = B$. The resampling factor M_i of subcarrier channel i is inversely proportional to the bandwidth of that channel, $M_i = B/B_i$, and thus an $L \times L$ DSXC could be built by using resampling filters with resampling factors $M_1, M_2, \ldots, M_L$. For a resampling FIR filter with a resampling factor M_i and a length of coefficients S, if the degree of parallelism in the DSP system is R, the number of required DSP slices to build this resampling FIR filter is $R \times S/M_i$. Since a DDC block needs a decimation BPF and a DUC block needs an interpolation BPF for each subcarrier channel, all together the DSXC needs L decimation BPF and L interpolation BPF. Therefore, the total DSP cost for building these resampling filters is

$$C_2 \approx \sum_{i=1}^{L} 2 \times R \times S/M_i = 2S \times R. \tag{2}$$

According to Eq. (2), the total DSP cost of DSXC based on resampling filters for frequency translation is independent of the number of subcarrier channels. Basically, a higher channel count requires a larger resampling factor for resampling filters, which reduces the DSP cost of each filter, and thus the total DSP cost does not increase with the number of channels. In comparison, for frequency translation based on the I/Q mixing and filtering technique, each digital filter requires the same amount of DSP slices, and thus the overall DSP cost increases linearly with the number of subcarrier channels.

The inset of Fig. 7, obtained through Eqs. (1) and (2), shows the DSP cost of DSXC based on two different methods. Here we assume the length of the filter coefficients is $S = 80$ and the degree of parallelism is $R = 4$. The DSP cost of the DSXC based on resampling filters remains unchanged when the number of subcarriers increases, whereas the DSP cost increases linearly with the number of subcarriers for I/Q mixing and filtering, and there will not be enough DSP slices available on a Xilinx Virtex-7 690T FPGA if the number of subcarriers exceeds five.

Although we used 4 × 4 DSXC as the example with equal subcarrier channel spacing and equal data rate for each channel, unequal channel spacing and different bandwidths for the subcarrier channels can also be used because the resampling factor for each channel can be independently set. This has been experimentally demonstrated and will be discussed in the next section.

V. Experiments

In order to demonstrate DSXC and test its performance experimentally, an optical system based on digital subcarrier multiplexing has been set up using an FPGA platform for real-time DSP and cross-connect switching.

The experimental setup is shown in Fig. 8, where an arbitrary waveform generator (AWG) generates an electrical waveform that has multiple subcarriers. A linear optical transmitter converts this multicarrier electrical waveform into the optical domain through direct intensity modulation. The optical signal is transmitted through a 25 km standard single-mode fiber (SMF) and detected by an optical receiver that linearly converts the received optical

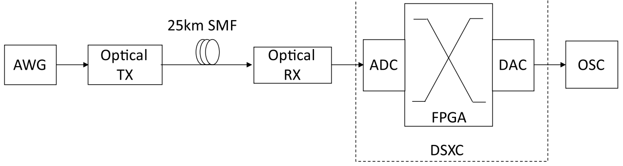


Fig. 8. Experimental setup.

signal into an electrical waveform. This detected electrical waveform is then sent into the DSXC for subcarrier-level cross-connect switching. The waveform at the output of the DSXC is sampled by an oscilloscope (OSC) for analysis.

This DSXC platform consists of three major parts: a FPGA board (Hitech-global HTG700), an ADC (TI ADC12J4000EVM), and a DAC (TI DAC38RF82EVM). The resolutions of the ADC and the DAC are 12 bits and 16 bits, respectively. Because a common clock is required, both the ADC and the DAC are running at an input sampling rate of 1.6 GSPS so that the available analog bandwidth is 800 MHz. The FPGA board is mounted with a Xilinx Virtex-7 690t FPGA chip. The FPGA clock frequency is 200 MHz, and thus the sampled data in the FPGA is processed in eight parallel channels.

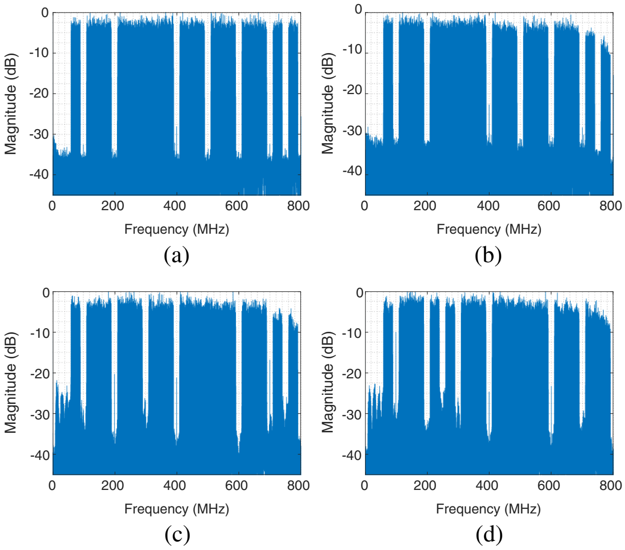
Based on this DSP platform, an 8 × 8 DSXC has been implemented, which switches subcarrier channels with three different data rates. Only resampling filters have been used for frequency translation in the experiment. The available analog bandwidth of 800 MHz is divided into eight frequency slots with three different widths: 200, 100, and 50 MHz, with 20 MHz reserved for the guard band between adjacent subcarrier channels. Thus, the bandwidths of the corresponding subcarrier channels are 180, 80, and 30 MHz, respectively, and in principle each can have an independent modulation format. Equiripple 108th-order FIR filters are used in this experiment for DDC and DUC. The ripple in the filter passband is 0.5 dB, and the stopband attenuation is 30 dB.

Table I shows an example of the input signal to the 8 × 8 DSXC, in which eight subcarrier channels, SC1, SC2, … SC8, are generated by the AWG. To demonstrate the capability of working with mixed modulation formats and data rates, Table I shows the bandwidth and modulation format assignment for the eight subcarrier channels.

Figure 9(a) shows the spectrum of the signal at the output of the optical receiver, where each subcarrier channel has almost equal amplitude. In order to characterize the effects of this DSP platform imposed on the signal, we measured the output of the DSP platform without cross-connect switching, and the spectrum is shown in Fig. 9(b). Although most channels from SC1 to SC6 have nearly the same amplitude at the output for the frequency range of <700 MHz, high-frequency channels SC7 and SC8 experience large roll-off of more than 5 dB for the frequencies beyond 700 MHz. This roll-off is mainly caused by ADC and DAC.

TABLE I
AWG-Generated Input Signal to 8 × 8 DSXC

Subcarrier #	Bandwidth (MHz)	Modulation Format
1	30	QPSK
2	80	QPSK
3	180	16QAM
4	80	16QAM
5	80	QPSK
6	80	QPSK
7	30	QPSK
8	30	QPSK



(a)

(b)

(c)

(d)

Fig. 9. Spectra of output signal: (a) at optical receiver output, which is DSXC input, (b) after the DSP platform but without switching, (c) after DSP platform with switching assignment of DSXC1, (d) after the DSP platform with switching assignment of DSXC2.

Figures 9(c) and 9(d) show the spectra after subcarrier switching for two different output channel assignments. In Fig. 9(c), denoted as DSXC1, the original subcarrier channels [1 2 3 4 5 6 7 8] have been switched to [7 4 6 5 3 2 8 1] at the output, while for the spectrum shown in Fig. 9(d), denoted as DSXC2, the original subcarrier channels [1 2 3 4 5 6 7 8] are switched to [8 6 1 7 2 3 4 5].

To evaluate the impact of DSXC on the signal quality, the waveforms at the DSXC input and output are processed to find the error vector magnitude (EVM) for each subcarrier channel. As the frequency response of the DSP platform, including ADC and DAC, is deterministic as shown in Figs. 9(b)–9(d), its impact can be digitally compensated for in frequency domain at the transmitter and/or the receiver.

Figure 10 shows the EVM of the eight subcarrier channels in four different scenarios. Open squares show the EVM measured at the input of DSXC, which is after 25 km fiber transmission and detected by the optical receiver. Open circles show the EVM after passing through the DSXC platform but without cross-connect switching, and thus no digital filters are applied for each subcarrier channel. The EVM degradation compared to those shown by open squares is primarily due to the frequency-dependent transfer functions and high-frequency roll-offs of ADC and DAC. Although we have applied slope compensation at the receiver, a small amount of EVM degradation still exists, especially for high-frequency channels. Open triangles in Fig. 10 show the EVM values of all channels after cross-connect switching with two different output channel assignments corresponding to the spectra shown in Figs. 9(c) and 9(d). The additional EVM degradation compared to those without switching is mainly attributed to resampling filters. This includes the impact of passband

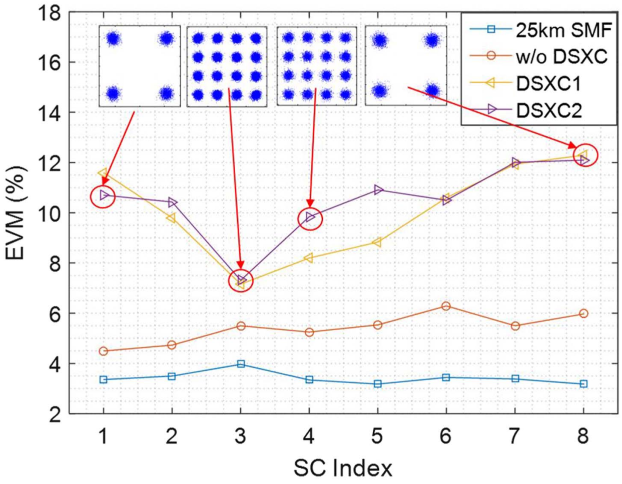


Fig. 10. Signal EVM of recovered subcarrier channels.

ripple, which directly contributes to the increase of EVM and the inter-subcarrier crosstalk because of the insufficient stopband attenuation. Several representative constellation diagrams are shown in the inset of Fig. 10, including channels with both quadrature phase-shift keying (QPSK) and 16QAM modulation, and at different frequency slots.

The FPGA resources used to build this DSXC are summarized in Table II. This includes 56.97% of the DSP slice usage, which is often the bottleneck for this application. There would not be enough DSP slices available with a Xilinx Virtex-7 690t to build this DSXC if the I/Q mixing and filtering method was used.

As mentioned in Section IV, higher-order digital filters help reduce passband ripple and increase stopband attenuation. FIR filters with lower ripple in the passband and higher suppression in the stopband would result in less EVM degradation. In fact, the EVM percentage of a signal constellation diagram is monotonically increased with the passband ripple. However, increasing the order of digital filters would increase the FPGA resource cost, especially the DSP cost. In addition, higher-order digital filters would also introduce longer processing delays for the signal.

In terms of signal processing delay, the latency of this DSXC is mainly introduced by the FIR filters. In fact, the latency of a FIR filter is $t_L = T_C(S-1)/2$, where $S-1$ is the filter order with S as the length of the filter coefficients, and T_C is the clock period. Latency caused by other utility logic such as cross-bar switches and data-type converters is only a few clock periods, which is negligible compared to t_L. With a clock period of $T_C = 5$ ns, the latency of each FIR filter as the function of the FIR filter order is shown in Fig. 11. By sending an impulse to the DSXC in the simulation based on the Xilinx system generator, the overall latency of DSXC in this system, including two Equiripple 108th-order FIR filters and other utility logics, was found to be less than 0.65 μs.

TABLE II
FPGA Resource Cost of DSXC

Resource	Utilization	Available	Utilization (%)
LUT	38909	433200	8.98
LUTRAM	18929	174200	10.87
FF	60475	866400	6.98
BRAM	96	1470	6.53
DSP	2051	3600	56.97

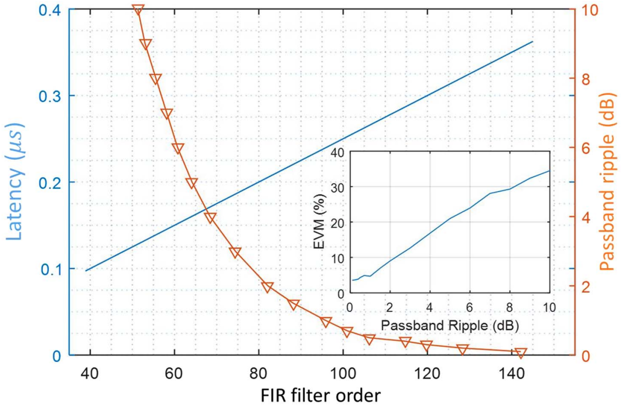


Fig. 11. Effects imposed by the FIR filter.

The simulation result also shows the impact of filter order on the EVM of the received signal. In the simulation, a subcarrier channel with QPSK signal is selected by an Equiripple FIR filter, and EVM is calculated as the function of filter order, as shown in Fig. 11. With the increase of filter order, the passband ripple decreases, and thus EVM improves. The inset of Fig. 11 shows the calculated relation between passband ripple and the signal EVM, which indicates that the EVM increases monotonically with the increase of passband ripple.

VI. Conclusion

We described a real-time DSP-enabled 8 × 8 DSXC test bed implemented on a Virtex-7 FPGA platform. The functionality and performance of the 8 × 8 DSXC test bed are assessed in terms of signal quality, required FPGA resources, and cross-connect data plane latency. Frequency translation of individual subcarrier channels while being routed through the DSXC is achieved through digital resampling filters implemented on the FPGA. This solution reduces the required FPGA resources when compared to the more conventional I/Q mixing and filtering. To implement I/Q mixing and filtering, the amount of required FPGA resources increases linearly with the number of subcarrier channels, while it remains constant when using digital resampling filters.

The experimental results show that the 8 × 8 DSXC test bed successfully switches the spectral location of each individual subcarrier channel while it is routed through the DSXC. Each subcarrier channel can be independently assigned a specific bandwidth, modulation format, and position in the spectrum. In addition, the circuit-switching DSXC introduces a deterministic and relatively small delay (<1 μs) in the data plane compared to the hard-to-predict delay and jitter of commercial packet switches, which depend on the link utilization and packet size.

Due to its fine bandwidth granularity and high spectral efficiency (which cannot be achieved by today's optical cross-connects), DSXCs are suited for access and metro-area networks that support applications with stringent network round-trip time requirements, like 5G, cloud-assisted robotics, tele-surgery, and real-time gaming. For example, with its capability to mitigate transmission impairments in the digital domain, offer bandwidth flexibility, and support multiple modulation formats, DSXC represents a valid solution to concurrently support and switch a variety of RoF channels in the mobile network fronthaul. Applications of such capabilities have also been discussed for DSP-based analog RoF systems [14–16].

Appendix A

A traditional technique for frequency translation is through I/Q mixing and filtering. As shown in Fig. 2, in the DDC process, a DDS simultaneously generating sine and cosine waveforms is used to provide a pair of LOs. Assume the incoming signal data sequence on the ith subcarrier channel is $I_i(t) = A_i(t)\cos(2\pi f_i t + \varphi_i)$, where $A_i(t)$ is modulated amplitude, f_i is the carrier frequency, and φ_i is the carrier phase; the in-phase (I) and quadrature (Q) components of the LO are $\cos(2\pi f_i t)$ and $\sin(2\pi f_i t)$, respectively. After down-conversion mixing and low-pass filtering (LPF), the I and Q components of the baseband signal are $\frac{1}{2}A_i(t)\cos(\varphi_i)$ and $-\frac{1}{2}A_i(t)\sin(\varphi_i)$, respectively. If the subcarrier channel needs to be dropped at this node, the I and Q components are combined together to recover the original baseband signal. Otherwise, the I and Q components are mixed with another pair of LOs, $\cos(2\pi f_j t)$ and $\sin(2\pi f_j t)$, in the DUC module. The frequency up-conversion generates $\frac{1}{2}A_i(t)\cos(\varphi_i)\cos(2\pi f_j t)$ and $-\frac{1}{2}A_i(t)\sin(\varphi_i)\sin(2\pi f_j t)$, and they are combined to form the DUC module output as

$$O_m(t) = \frac{1}{2}A_i(t)\cos[2\pi f_j t + \varphi_i]. \tag{A1}$$

Throughout this frequency-translation process, the carrier frequency is changed from f_i to f_j, while the original carrier phase, φ_i, is automatically maintained.

Note that I/Q mixing requires two separate filters for the I and the Q channels. Alternatively, one can use a single-stage frequency translation from f_i to f_j using an LO frequency $|f_j - f_i|$. However, to avoid spectral overlap with other subcarrier channels, this approach requires the selection of the subcarrier channel at f_i by a BPF before mixing, and the selection of subcarrier frequency at f_j by another BPF after mixing. Thus, the number of digital filters remains unchanged.

Alternatively, frequency translation may be applied to complex field modulated subcarriers in which the upper and lower sidebands of each subcarrier channel are not redundant. In that case, a Hilbert transform must be applied to avoid spectral aliasing, which would further increase the DSP cost.

Appendix B

Digital frequency translation can also be accomplished with resampling and filtering. As shown in Fig. 3(a), DDC can be achieved by applying a BPF before a down-sampling unit so that the subcarrier channel selected by the BPF is down-converted to a lower frequency slot. Let Q be the down-sampling factor, indicating that one of every Q output samples from the BPF is retained, while the other $Q - 1$ samples are discarded. The processing of these $Q - 1$ discarded samples is in fact unnecessary and could be avoided to reduce the DSP cost. By combining down-sampling and filtering into a single decimation filter as shown in Fig. 3(b), only one of every Q sampling operations is actually performed for BPF. Therefore, the total computation is effectively reduced by a factor of Q, and the DSP cost of a decimation BPF is only $1/Q$ of a conventional BPF with the same number of coefficients.

Similarly, DUC can be achieved by using an up-sampling unit followed by a BPF, as shown in Fig. 3(c), so that a subcarrier channel at a lower frequency slot is up-converted to a higher frequency slot and selected by the BPF. For an up-sampling factor of P, $P - 1$, zeros are inserted between every two samples of the input digital sequence in the up-sampling process. As the multiplication of these inserted zeros with filter coefficients always results in zeros in the subsequent digital-filtering process, these operations are not necessary. By combining up-sampling and filtering into a single interpolation filter as shown in Fig. 3(d), unnecessary operations performed on the inserted zeros can be avoided by using only $1/P$ of the BPF coefficients during each convolution. Thus, the total computation is effectively reduced by a factor of P, and the DSP cost of the interpolation BPF is only $1/P$ that of a traditional BPF with the same number of coefficients.

The frequency translation based on resampling filters and its DSP cost can be analyzed mathematically. In Fig. 3(a) the input sequence $x[n]$ represents a digital multi-carrier signal. Suppose that the BPF is an N-tap FIR filter with coefficients $h_0, h_1, \ldots, h_{N-1}$, then the filter output is

$$y[n] = \sum_{i=0}^{N-1} h_i \cdot x[n-i], \tag{B1}$$

which is the digital sequence of a selected subcarrier channel that needs to be down-converted. After down-sampling by a factor of Q, the output is

$$z[n] = y[nQ]. \tag{B2}$$

The discrete-time Fourier transform (DTFT) of the down-sampled digital sequence $z[n]$ is

$$Z(e^{j\Omega}) = \frac{1}{Q}\sum_{p=0}^{Q-1} Y(e^{j\frac{\Omega-2\pi p}{Q}}), \tag{B3}$$

where $Y(e^{j\Omega})$ is the DTFT of $y[n]$.

According to Eq. (B3), $Z(e^{j\Omega})$ is an expanded and shifted version of $Y(e^{j\Omega})$ with an expansion factor Q. Since the DTFT of a digital sequence is periodical with a period of 2π, every spectral component with an original bandwidth π/Q will be expanded to π. As a result, the original spectrum, which occupies a frequency slot (FS) with a bandwidth of π/Q, will be automatically expanded to π, and frequency shifted by $2p\pi$ into Q copies with $p = 0, 1, 2\ldots Q-1$.

Although $Y(e^{j\Omega})$ is continuous and periodical, the digital Fourier transform (DFT) of $y[n]$ is one sampled period of its DTFT, which is sampled at discrete points $\Omega = 2\pi k/M$, where Ω is a normalized angular frequency, M is the length of DFT, and $k = 0, 1, 2\ldots M-1$ is the index of sampling in the frequency domain. Thus, $\Omega \in (0, 2\pi)$ for $M \gg 1$. If the sampling rate of $y[n]$ is F_s, the actual frequency range of its DFT, denoted by $Y(f)$, is $(-F_s/2, F_s/2)$.

As described previously, it is unnecessary to calculate the values of the samples in $y[n]$ that are not used in the subsequent down-sampling process. The BPF and the down-sampling unit could be more efficiently implemented together as a decimation filter as shown in Fig. 3(b). Considering Eqs. (B1) and (B2), the output of the decimation BPF is

$$z[n] = \sum_{i=0}^{N-1} h_i \cdot x[nQ-i]. \tag{B4}$$

Compared with Eq. (B1), the amount of calculations in Eq. (B4) has been reduced by a factor of Q. This is because the number of output samples of the decimation filter is Q times less than that of a traditional digital filter, and thus, the DSP cost is reduced by a factor of Q.

Similarly, for the conventional up-sampling process shown in Fig. 3(c), suppose the input digital sequence is $x[n]$; after up-sampling by a factor of P, the output is

$$y[n] = \begin{cases} x[\frac{n}{P}] & \text{if } n/p \text{ is an integer} \\ 0, & \text{otherwise} \end{cases}. \tag{B5}$$

The DTFT of the up-sampled digital sequence $y[n]$ is

$$Y(e^{j\Omega}) = X(e^{j\Omega P}), \tag{B6}$$

where $X(e^{j\Omega P})$ is the DTFT of $x[n/P]$. In this up-sampling process, $Y(e^{j\Omega})$ is a compressed version of $X(e^{j\Omega})$, and the compression factor is equal to the up-sampling factor P.

This up-sampling process can be explained by the similar scaling rule between DFT and DTFT as described above for down-sampling. An up-sampling by a factor P is equivalent to creating P equally spaced copies of the DFT of $x[n]$, denoted by $Y(f)$, in the expanded frequency range of $(-PF_s/2, PF_s/2)$.

As shown in Fig. 3(c), $y[n]$ and $z[n]$ are the input and output of the BPF, respectively. Suppose this BPF is a N-tap FIR filter with coefficients $h_0, h_1, \ldots, h_{N-1}$, then the output of this BPF is

$$z[n] = \sum_{i=0}^{N-1} h_i \cdot y[n-i]. \tag{B7}$$

As described previously, the up-sampling unit and BPF can be combined into an interpolation BPF to reduce the DSP cost. Considering Eqs. (B5) and (B7), the output of this combined interpolation BPF is

$$z[n] = \sum_{k=0}^{N/P-1} h_{n-kP} \cdot x[k]. \tag{B8}$$

Compared with Eq. (B7), Eq. (B8) only requires N/P instead of N multiplications so that the DSP cost is reduced accordingly by a factor of P.

Acknowledgment

This work was supported in part by the U.S. National Science Foundation under Grant CNS-1409853 and Grant CNS-1405405.

References

[1] O. Gerstel, M. Jinno, A. Lord, and S. B. Yoo, “Elastic optical networking: a new dawn for the optical layer?” *IEEE Commun. Mag.*, vol. 50, no. 2, pp. s12–s20, 2012.

[2] K. Christodoulopoulos, I. Tomkos, and E. A. Varvarigos, “Elastic bandwidth allocation in flexible OFDM-based optical networks,” *J. Lightwave Technol.*, vol. 29, no. 9, pp. 1354–1366, 2011.

[3] K. Christodoulopoulos, I. Tomkos, and E. A. Varvarigos, “Dynamic bandwidth allocation in flexible OFDM-based networks,” in *Optical Fiber Communication Conf.*, Optical Society of America, 2011, paper OTuI5.

[4] M. Jinno, B. Kozicki, H. Takara, A. Watanabe, Y. Sone, T. Tanaka, and A. Hirano, “Distance-adaptive spectrum resource allocation in spectrum-sliced elastic optical path network [topics in optical communications],” *IEEE Commun. Mag.*, vol. 48, no. 8, pp. 138–145, 2010.

[5] P. M. Hill and R. Olshansky, “A 20-channel optical communication system using subcarrier multiplexing for the transmission of digital video signals,” *J. Lightwave Technol.*, vol. 8, no. 4, pp. 554–560, 1990.

[6] R. Hui, B. Zhu, R. Huang, C. T. Allen, K. R. Demarest, and D. Richards, “Subcarrier multiplexing for high-speed optical transmission,” *J. Lightwave Technol.*, vol. 20, no. 3, pp. 417–427, 2002.

[7] Y. Zhang, M. O’Sullivan, and R. Hui, “Digital subcarrier multiplexing for flexible spectral allocation in optical transport network,” *Opt. Express*, vol. 19, no. 22, pp. 21880–21889, Oct. 2011.

[8] G. Bosco, A. Carena, V. Curri, P. Poggiolini, and F. Forghieri, “Performance limits of Nyquist-WDM and COOFDM in high-speed PM-QPSK systems,” *IEEE Photon. Technol. Lett.*, vol. 22, no. 15, pp. 1129–1131, 2010.

[9] R. Hui, W. Huang, Y. Zhang, M. Hameed, M. Razo, M. Tacca, and A. Fumagalli, "Digital subcarrier cross-connects (DSXCs)," in *14th Int. Conf. on Transparent Optical Networks (ICTON)*, IEEE, 2012, pp. 1–6.

[10] R. Hui, W. Huang, Y. Zhang, M. Hameed, M. Razo, M. Tacca, and A. Fumagalli, "Digital subcarrier optical networks and cross-connects," *J. High Speed Netw.*, vol. 19, no. 1, pp. 55–69, 2013.

[11] A. Pizzinat, P. Chanclou, F. Saliou, and T. Diallo, "Things you should know about fronthaul," *J. Lightwave Technol.*, vol. 33, no. 5, pp. 1077–1083, 2015.

[12] X. Liu and F. Effenberger, "Emerging optical access network technologies for 5G wireless," *J. Opt. Commun. Netw.*, vol. 8, no. 12, pp. B70–B79, 2016.

[13] D. Novak, R. B. Waterhouse, A. Nirmalathas, C. Lim, P. A. Gamage, T. R. Clark, M. L. Dennis, and J. A. Nanzer, "Radio-over-fiber technologies for emerging wireless systems," *IEEE J. Quantum Electron.*, vol. 52, no. 1, pp. 1–11, 2016.

[14] D. A. A. Mello, A. N. Barreto, F. A. Barbosa, C. Osorio, M. Fiorani, and P. Monti, "Spectrally efficient fronthaul architectures for a cost-effective 5G C-RAN," in *18th Int. Conf. on Transparent Optical Networks (ICTON)*, IEEE, 2016, pp. 1–5.

[15] X. Liu, H. Zeng, N. Chand, and F. Effenberger, "Efficient mobile fronthaul via DSP-based channel aggregation," *J. Lightwave Technol.*, vol. 34, no. 6, pp. 1556–1564, 2016.

[16] X. Liu, H. Zeng, and F. Effenberger, "Bandwidth-efficient synchronous transmission of I/Q waveforms and control words via frequency-division multiplexing for mobile fronthaul," in *Global Communications Conf. (GLOBECOM)*, IEEE, 2015.

[17] "IEEE Standard for Local and metropolitan area networks–Bridges and bridged networks," IEEE Std. 802.1Q-2014 (Revision of IEEE Std 802.1Q-2011), Dec. 2014, pp. 1–1832.

[18] C. Kachris and T. Ioannis, "A survey on optical interconnects for data centers," *IEEE Commun. Surv. Tutorials*, vol. 14, no. 4, pp. 1021–1036, 2012.

[19] 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*, vol. 20, no. 1, pp. 317–337, 2012.

[20] 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*, vol. 20, no. 26, pp. B543–B551, 2012.

[21] W. Jin, X. Duan, Y. Dong, B. Cao, R. P. Giddings, C. Zhang, K. Qiu, and J. M. Tang, "DSP-enabled flexible ROADMs without optical filters and OEO conversions," *J. Lightwave Technol.*, vol. 33, no. 19, pp. 4124–4131, 2015.

[22] R. P. Giddings, E. Al-Rawachy, and J. M. Tang, "Experimental demonstration of real-time add/drop operations in DSP-enabled flexible ROADMs for converging fixed and mobile networks," in *Optical Fiber Communication Conf.*, Optical Society of America, 2018, paper W2A.33.

[23] F. J. Harris, *Multirate Signal Processing for Communication Systems* Prentice Hall PTR, 2004, Chap. 2.

[24] Xilinx, "FIR compiler V7.2 logiCORE IP product guide," Nov. 2015 [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/fir_compiler/v7_2/pg149-fir-compiler.pdf.

[25] F. J. Harris, C. Dick, and M. Rice, "Digital receivers and transmitters using polyphase filter banks for wireless communications," *IEEE Trans. Microwave Theory Tech.*, vol. 51, no. 4, pp. 1395–1412, 2003.

[26] Xilinx, "Virtex-7 product table," May 2015 [Online]. Available: https://www.xilinx.com/support/documentation/selection-guides/virtex7-product-table.pdf.