Low-Power SDR Design on an FPGA for Intersatellite Communications

Xin Cai, Mingda Zhou, Tian Xia, Senior Member, IEEE, Wai H. Fong, Wing-Tsz Lee, and Xinming Huang, Senior Member, IEEE

Abstract—Small satellite systems make space missions for communication, navigation, and scientific research more realizable and diversified. Small satellites flying in large clusters or constellation formation as a network can provide an economical access to accomplish more complex missions, such as distributed computation, high-resolution imaging, and spacecraft maintenance. An increasing number of satellites operating on lower earth orbit for complex missions require a wireless communication system that is both reliable and flexible. This paper presents a complete software-defined radio (SDR) model for intersatellite communications (ISCs) and its implementation on a field-programmable gate array (FPGA). The proposed SDR for transmitter and receiver only has a power consumption of 2.1 and 3.2 W, respectively, which is suitable for power-limited small satellite systems. Algorithms and parameters of each block are optimized aiming at reducing hardware resource utilization. A low-density parity-check code constructed by the Euclidean geometry method is adopted as the channel code for forward error correction. Implementations of the synchronization, demodulation, and decoding algorithms are optimized for hardware efficiency. The low-power SDR designs are implemented on an FPGA-based experimental platform and successful demonstrated by over-the-air transmissions.

Index Terms—Field-programmable gate array (FPGA), intersatellite communications (ISCs), low-density parity-check (LDPC) code, software-defined radio (SDR), synchronization.

I. INTRODUCTION

Small spacecraft has attracted many interests for launching scientific missions at or beyond lower earth orbit in recent years. Small satellites are becoming an important part in small spacecraft missions owing to their advantages of low cost, small weight, short lead time, and miniature size [1]. Hundreds of small satellites flying together are able to accomplish more missions, e.g., higher precision navigation, formation control, data exchange, information processing, and spacecraft maintenance [2]. To integrate the processing capability, software and hardware resources of a single satellite, and to keep a group of formation-flying satellites work efficiently, all the satellites have to communicate with each other. Thus, intersatellite communications (ISCs) with real-time processing are required to make a large number of small satellites that work as a network [3], [4]. A communication system design with high data rate, low bit error rate (BER), and real-time transmission is a prerequisite for an efficient ISC.

The open system interconnection (OSI) and the derivative models are usually adopted to design the ISCs’ structure. Frequency allocation, data rate, modulation, code schemes, and platform are the main considerations in the OSI physical layer. The choice of a frequency band for ISCs is limited by several aspects: spectrum allocated by the International Telecommunication Union, the available hardware, power supply, antenna size, and mission requirements. Typically, frequency allocation candidates for ISC include S-, Ku-, and Ka-bands [5], [6]. It is also known that a higher frequency for ISC not only helps to reduce the antenna size and mass of the transceiver but also makes a wider bandwidth available [7]. Binary phase-shift keying (BPSK), quadrature phase-shift keying (QPSK), and offset QPSK (OQPSK) are preferred in the ISC. A coherent BPSK system needs less power at the same BER level and provides higher sensitivity for communication and tracking. QPSK can enhance the spectral efficiency by carrying 2-bit information in each symbol, and OQPSK overcomes a sudden phase change problem in QPSK.

There are several research works on ISC in the existing literature, mostly focused on theoretical studies and numerical simulations. A Ka-band test bed was developed in [8], and a rate 1/2 convolution code with a constraint length of 7 was implemented with the OQPSK modulation in the downlink and the BPSK modulation in the uplink. However, their test results have shown that this test bed in a convolution code scheme could only achieve a BER of $10^{-4}$ at 5-dB bit energy-to-noise variance ($E_b/N_0$). In [9], a rate 1/2 low-density parity-check (LDPC) (512, 256) code was adopted to improve the BER performance. They demonstrated that this LDPC code had a better error correction capability than the convolution code. However, as many as 50 iterations were needed in the LDPC decoder.

Edmonson et al. [10] presented an integration of responsive and formal design (RFD) with an OSI framework for ISC.
Different levels of design abstraction for the RFD process are introduced. In [11], several methods were presented to improve a CPU and field-programmable gate array (FPGA) performance for satellite applications, and also the study described some potential software-defined radio (SDR) architectures to support signals for distributed satellite systems, but no design details for ISCs were illustrated. Based on the structure of small satellite networks (SSNs), Zhou et al. introduced the contact plan design problems [12]. The study was focused on SSNs to characterize the resources and to maximize the network data volume.

The SDR platform has been widely used in ISCs for its unique capabilities: technological flexibility, adaptive and multimode operations, programmable reconfiguration, and convenient upgrade [13], [14]. These capabilities allow the SDR platform to remotely change or upgrade the features of a radio to adapt to a dynamic environment and meet new requirements. CubeSat SDR can implement several communication protocols and support data rate of 10 Mb/s [15]. Researchers at the National Aeronautics and Space Administration (NASA) used the spectrum SDR-3000 platform as a test bed to simulate and demonstrate the inner satellites crosslink [16]. In [6], Universal Software Radio Peripheral N210 was employed with its transceiver operating in the frequency range from 400 MHz to 4.4 GHz. An AITech s950 single-board computer was integrated into the Harris SDR [8]. In [17], an FPGA-based SDR architecture was designed for the FUNcube-I(AO-73) CubeSat beacon signals on a platform with AD-FMCOMMS3 radio front end and ZedBoard FPGA back end.

In this paper, an SDR model for ISCs is designed and also implemented on the Zynq system-on-chip (SoC) FPGA-based SDR platform. The model is aimed at exploring an FPGA-based SDR architecture to provide an efficient data link for ISCs. The system model includes the entire ISCs components: coding, OQPSK modulation, and pulse shaping in the transmitter and automatic gain control (AGC), frequency compensation, timing recovery, frame synchronization, soft demodulation, and decoding in the receiver. The main contributions of this paper can be summarized as follows.

1) A complete SDR model and a system design for ISCs are proposed with performance simulation results. The proposed SDR architecture is model-based designed through using multiple tools and programming languages. Each individual component can be easily updated or replaced using a new module according to the targeted communication standard. The FPGA implementation of the transmitter and the receiver only consume the power of 2.1 and 3.2 W, respectively.

2) Different from the conventional LDPC codes encoding methods, we propose a new encoding scheme aiming at saving more hardware resources. Furthermore, an improved encoding algorithm and the corresponding encoder structure are also proposed. The improved algorithm helps to save hardware resources significantly by sharing the parity check bit generator.

3) A high code rate LDPC (255, 175) code is constructed with the Euclidean geometry method, which improves the BER performance significantly compared with the convolution code used in the Harris test bed and the LDPC code in [9]. A simplified LDPC decoder with a fully parallel structure is designed, in which an efficient three-level comparison tree is proposed to realize the check nodes (CNs) update. We also apply several optimization techniques that reduce the hardware usage by two thirds with only very small performance degradation.

4) A simplified magnitude detection scheme is proposed, and the optimal combination coefficients are found to reduce the error introduced by the simplification. An improved Luise algorithm is adopted for the coarse frequency compensation (CFC) aiming at reducing the computation complexities. An over-the-air transmission in the S-band is demonstrated, and a signal quality is measured during the real-time transmissions.

The remaining sections of this paper are organized as follows. The features of intersatellite links and some considerations for ISCs are discussed in Section II. Design strategies and related algorithms are illustrated in Section III. The structure of transmitter design is revealed in Section IV. Section V illustrates the implementation details of the receiver. The process of targeting the transceiver models on the SDR platform, implementation results, and so on are discussed in Section VI. Conclusions are made in Section VII.

II. FEATURES OF INTERSATELLITE COMMUNICATION LINKS AND DESIGN CONSIDERATIONS

This paper is aimed to design efficient ISCs links for a group of small satellites flying in cluster formation, in which 100~150 small satellites are deployed in a limited spherical space with a diameter of 100 m at an altitude of 500 km above the earth. These satellites are expected to be only of laptop size with the total mass less than 5 kg and the total power less than 10 W. Small latency is required so that ISCs links can be responded in a short time. It is known from [1] that the rechargeable battery system occupies about one-third weight of the entire satellite and the battery cell energy density ranges from 120 to 250 Wh/kg. For an individual satellite weighted less than 5 kg, the battery system cannot be heavier than 1.6 kg, which can provide a limited energy of about 192~400 Wh. Compared with traditional wireless communication systems, the ISCs system has stringent limitations on size, weight, and power. Each module in the system should be optimized for hardware and energy efficiency. Many issues need to be considered, including the limitation of modulation types, the choice of error correction code, the AGC to overcome signal fading, the compensation for Doppler frequency shift, and radiation-hardening devices.

When engineers design a satellite system, high-power traveling-wave tubes (TWTs) and solid-state power amplifiers are often utilized. To obtain the maximal amplification efficiency and maximize the output power, TWTs and solid-state power amplifiers are usually designed to work at or beyond the compression point, which may bring distortion to the original signals. The signals around the outer states of a constellation are more susceptible to distortion, since they need more power to transmit. The distortion caused by the amplifiers drives...
the received signals around the states to move toward the inner rings of a constellation, resulting in much difficulty during decoding. Therefore, the modulation schemes for small satellites are often very limited. Typical modulation candidates for ISCs include BPSK, QPSK, OQPSK, and amplitude phase shift keying.

The power limitation also has an effect on the choice of code length. A decoder for a longer code often requires more hardware resources, resulting in higher power consumption. In addition, a longer code may also increase the decoding latency. Therefore, a shorter code is often selected for ISCs. NASA’s first $Ka$-band transceiver implemented on Harris’s SDR adopted a rate 1/2 convolution code with a constraint length of 7 [8]. In this paper, the (255, 175) LDPC code is constructed, which has much better performance than the convolution code in Harris’s SDR and also meets the requirements of small decoding latency.

Free-space loss refers to the energy loss of electromagnetic wave when it travels in the air. An electromagnetic wave mainly travels in a free space beyond the atmosphere and the free-space loss is proportional to the square of carrier frequency and the square of the distance between two satellites [18]. The channel model for ISCs is mainly determined by the free-space loss and thermal noise from the electronics so that the channel can be assumed as additive white Gaussian noise [19]. To overcome the potential signal fading, an AGC module is designed to adjust the amplitude of the received signals automatically.

The Doppler effects in an orbiting cluster are expected to be insignificant, since all the satellites fly in a limited spherical space and the relative velocity between the satellites is small. The Doppler effect is dominated by the line of sight velocity differences. For intersatellite links working in the $Ka$-band, the maximum Doppler shift frequency is around a few hundreds of hertz. The Doppler shift frequency is smaller for satellites that operate in a lower frequency band. However, a frequency shift detector and a compensator are still needed. In this paper, a two-step frequency compensator is designed to compensate the frequency shift caused by the transmission and Doppler effect.

Single-event upset introduced by a cosmic ray has an effect on most commercial off-the-shelf devices. Semiconductor components easily suffer from the radiation damage. This drives many manufacturers to develop radiation-tolerance devices. Xilinx has already released a series of rad-hard and rad-tolerant FPGAs. These rad-tolerant FPGAs are robustly designed to meet the reliability requirements in space environment. The reconfigurability of an FPGA allows the modifications at any time and upgrades of system after launch without any hardware replacement, which greatly reduces the cost and risk of a space mission.

III. DESIGN STRATEGIES AND RELATED ALGORITHMS

The specification of laptop-size satellites determines that they must have these special features: low weight, extremely limited hardware resources, and limited power. Despite these limitations, a reliable link is still needed. A qualified transceiver for ISCs must meet these requirements very well.

A. Resources-Saving Strategies

For the same BER performance, the signal-to-noise ratio requirement can be significantly reduced with forward error correction (FEC) coding, which also leads to lower transmission power. An LDPC code is widely used in the latest communication standards, since its capacity can approach the Shannon limit quite closely.

In small satellite communications, a codeword with some special requirements is needed: a code rate is higher than 1/2 and the size of information block or code block should be an integer multiple of 8 or 16 [6] so that information block can be integral multiple of a byte. In this paper, a 2-D type-I $(0, 4)$th-order Euclidean geometry LDPC code with 175 information bits and 255 code bits, denoted as EG-LDPC (255, 175), is adopted [20]. Considering the OQPSK modulation adopted in this model, an extra bit 0 is added, which results in the code length of 256.

An $(n, k)$ LDPC code can be defined by its parity check matrix $H$. The $H$ matrix of the EG-LDPC (255, 175) is of size $255 \times 255$. The fact that the dimension of this matrix is only 80 makes it possible to simplify the matrix $H$. The 80 rows that are linearly independent can be found. Thus, the 80 rows can form a new parity check matrix $H'$. A traditional way to obtain the systematic codeword is to convert $H$ into the form of $[H^d, H^p]$ (denoted as $H_d$), where $H^p$ is a lower triangular matrix.

In this paper, we change $H$ into another special form (denoted as $H_d$) to implement the encoding so that some benefits and different characters can be obtained. The submatrix $H^p$ is a “dual-diagonal” matrix, which can be expressed as

$$H^p = \begin{pmatrix} 1 & & & & & & & \ & 1 & & & & & & \\ & & \ddots & & & & & \\ & & & & \ddots & & & \\ & & & & & 1 & & \\ & & & & & & \ & & & 1 & \\ \end{pmatrix}. \tag{1}$$

The encoding can be realized by implementing the following recursion (set $p_0 = 0$):

$$p_j = p_{j-1} \oplus \left( \bigoplus_{i \in B_j} d_i \right), \quad j = 1, 2, \ldots, (n - k) \tag{2}$$

where $d_i$ is the $i$th information bit, $p_j$ is the $j$th parity check bit, and $B_j = \{ i : H_d^p, i = 1 \}$. An example of the encoding process is shown in Fig. 1.

The benefits of encoding based on this “dual-diagonal” form can be illustrated from three aspects. First, the codes that are defined by $H_d$ ($H^p$ is of the “dual-diagonal” form) constitute

![Fig. 1. Example of the encoding process.](image-url)
There are many LDPC codes, such as repeat accumulate (RA) codes [21], semirandom LDPC (SRLDPC) [22], and concatenated tree (CT) codes [23], which have the same $H^p$ matrix. Thus, this subclass of an LDPC code may have the same features with RA, SRLDPC, and CT codes. Second, the “dual-diagonal” form makes the hardware implementation more flexible and contributes to saving hardware resources. When the parity check bit $p_j$ is calculated, designers only need the value of the former parity check bit $p_{j-1}$ and the values of the related information nodes (information nodes between two black nodes $p_{j-1}$ and $p_j$ in Fig. 1). In addition, there are only two “1” in each row of $H^p$ except the first row. Compared with traditional $H_i$ with a lower triangular matrix, fewer XOR operations are required when the parity check bits are calculated. Thus, fewer XOR units are needed in hardware implementation, which helps to save hardware resources. Finally, fewer computations are required in the encoding process. There are $(n^2 - k^2)/2$ computations in the traditional encoding method. However, there are at most $(k + 2)(n - k)$ computations needed in the encoding process based on $H_d$.

Although the improvements from the above-described encoding procedure help to reduce the XOR operations, there are still 80 groups of XOR operations needed when the encoder for EG-LDPC (255, 175) is implemented. Motivated by the module-sharing technology, we further propose a new architecture in which the parity check bit generator is shared. There is only one group of XOR operation required in the new structure. The details of the structure will be illustrated in Section IV.

Since the $H$ matrix of this code is of size $255 \times 255$, typical decoding algorithms, such as sum-product algorithm (SPA) and min-sum algorithm (MSA), indicate that a conventional decoder with a fully parallel structure requires 255 check note processors (CNPs) and 255 variable node processors (VNPs). To reduce the hardware resources, a simplified decoder is proposed and implemented. The proposed decoder is implemented based on the new parity check matrix $H^s$ so that only 80 CNPs and 255 VNPs are required, which results in fewer usage of 175 CNPs.

Before the radio-frequency (RF) signals are fed into FPGA, an AGC module is necessary to regulate the amplitude of signals. The amplitude detection is involved in the square root operations. Conventional methods mainly adopt the intellectual property (IP) cores, lookup table (LUT) or coordinate rotation digital computer (CORDIC) algorithm. Since the complex signals of RF modules entering the FPGA are distributed in a limited area around the constellation points, an alternative simplification method is proposed, in which the amplitude is estimated by the linear combination of the in-phase and quadrature components and only one adder is required.

**B. Building Reliable Links’ Strategies**

Reliable links are required to ensure the communications among satellites. The receiver should have good capabilities of correcting the frequency offset and the timing error and finding the frame header.

A frequency compensation module is needed to correct the frequency offset introduced by the transmission and Doppler effect. This module is typically realized by two methods: fine frequency compensation (FFC) and CFC followed by FFC. Although FFC may require fewer hardware resources, it only provides reliable compensation within a very narrow range of frequency offset. The adoption of CFC followed by the FFC can ensure a low BER for a wider range of frequency offset. The performance comparison of the two schemes is shown in Fig. 2. The comparison indicates that two-step frequency compensation (CFC followed by the FFC) is able to handle a wide range of frequency offset.

In this paper, a two-step frequency compensator is designed to compensate the signal frequency shift and Doppler frequency shift. The frequency compensator is implemented through obtaining the information from the fourth power of the received signal. This helps to save hardware resources compared with the fast Fourier transform (FFT)-based method. The mathematical expression of the received signals can be $e^{j(k\pi/2 + \Delta f + \Delta \phi)}$, where $k$ is an integer that ranges from 0 to 3, $\Delta f$ is the frequency offset, and $\Delta \phi$ is the phase offset. In the first stage, an FFC module estimates the frequency offset $\Delta f^\prime$. In the next stage, an FFC module estimates the residual frequency offset $\Delta f''$ and the phase offset $\Delta \phi^\prime$. Once $\Delta f'$, $\Delta f''$, and $\Delta \phi'$ are obtained, the signals can be recovered as $e^{j(k\pi/2 + \Delta f + \Delta \phi - \Delta f' - \Delta f'' - \Delta \phi')}$.

The precise symbol timing must be known in order to demodulate the symbols. In this paper, a zero-crossing timing error detector (ZCTED) is adopted to calculate the timing error and the half symbol timing offset introduced in the OQPSK modulation. The ZCTED is designed by finding the zero crossings in the eye diagram [24]. The timing error is fed into a loop filter and then goes through the interpolation control block. The interpolation control block calculates the timing difference between the actual sampling instant and the optimal instant. Interpolants are generated by the interpolation filter at the optimum sampling instants.

The frame synchronization module is used to find the start of a frame and locate the beginning of the payload. A frame synchronization module that has low false-detection and misdetection probability is required. Typically, a pilot is
adopted to facilitate the header detection. In this paper, a 13-bit Barker code is used as the pilot, since this Barker code is a maximum autocorrelation sequence with sidelobes not larger than one [25]. The optimal autocorrelation properties of the Barker code enable the system to find the header of a frame with high confidence.

C. Related Algorithms

Since the new matrix $H_d$ is adopted to conduct the encoding, some modifications are made compared with the traditional encoding algorithm. The pseudocode for the entire encoding procedure is given in Algorithm 1, which also gives further explanation to 2.

Algorithm 1 Encoder Based on $H_d$

**Input:** information bits vector: $(d_1, d_2, \ldots, d_k)$ and $H_d$

**Output:** codeword: $C$

1. allocate two pieces of storage $D$ and $P$ with the length of $k$ and $(n-k)$ respectively, set $D = 0$, $P = 0$;
2. for $l = 1$ to $n$ do
   1. if $l \leq k$ then
      1. $D[l] = d_l$;
      2. $C[l] = d_l$;
   2. else
      1. $P[m] = P[m - 1] \oplus \left( \bigoplus_{i \in B_m} D_i \right)$;
      2. $m = m + 1$;
      3. update $B_m$;
      4. $C[l] = P[m]$;
3. goto line 12;
4. $P[m] = P[m - 1] \oplus (F(Temp, d_{max}))$;
5. $C[l] = P[m]$;
6. $Temp = \{D_i : i \in B_m\}$;
7. goto line 5;

Although fewer XOR operations are needed in this algorithm, it can be found that $(n-k)$ groups of XOR operations are still required. Thus, we further improve the encoding algorithm to reduce the group numbers of XOR operations. Assume that $Temp$ is a bit vector with the length of $b$ and $F(Temp, b)$ is a function that returns the XOR value of the $b$ elements in the vector $Temp$. The improved encoding procedure is illustrated in Algorithm 2.

In the improved algorithm, a group of XOR operations $F(Temp, d_{max})$ is shared for $(n-k)$ times. When a certain parity check bit is generated, designers only need to update the $Temp$ vector. The updating process of the vector $Temp$ is equal to finding all the bits in the set $\{d_i : i \in B_m\}$.

The SPA has a powerful capability of correcting the bit errors for LDPC codes. However, a large amount of complex computations prevents the adoption of SPA in hardware implementation. The MSA is an ideal replacement to simplify the SPA due to the linear complexity and easier implementation of the CNs’ update. The major operation of the CNs’ update process in the MSA is equal to finding the minimal value from all the neighbors of a CN. However, successive comparisons to obtain the minimal value require a lot of clock cycles, which also results in long decoding latency. This must be avoided in the ISCs. Considering the new matrix $H'$ of this

Algorithm 2 Improved Encoder Based on $H_d$

**Input:** information bits vector: $(d_1, d_2, \ldots, d_k)$ and $H_d$

**Output:** codeword: $C$

1. allocate two pieces of storage $D$ and $P$ with the length of $k$ and $(n-k)$ respectively, set $D = 0$, $P = 0$;
2. find the maximal row degree $d_{max}$ of $H_d$;
3. allocate a piece of storage $Temp$ with the length of $d_{max}$, set $Temp = 0$;
4. $l = 1$, $m = 1$, $B_m = \{i : H_{m,i} = 1\}$;
5. for $l = 1$ to $n$ do
   1. if $l \leq k$ then
      1. $D[l] = d_l$;
      2. $C[l] = d_l$;
   2. else
      1. $Temp = \{D_i : i \in B_m\}$;
      2. $P[m] = P[m - 1] \oplus (F(Temp, d_{max}))$;
      3. $C[l] = P[m]$;
      4. $Temp = \{D_i : i \in B_m\}$;
      5. goto line 12;
6. $P[m] = P[m - 1] \oplus (F(Temp, d_{max}))$;
7. $C[l] = P[m]$;
8. $Temp = \{D_i : i \in B_m\}$;
9. goto line 5;

EG-LDPC (255, 175) code still has a regular row degree of 16, we designed a three-level comparison tree that can find the minimal and second minimal values in only one clock cycle. More details about the MSA are introduced in [26].

In the frequency compensation module, FFT-based and correlation-based algorithms (also known as the Luise algorithm [27]) are the common methods to estimate the frequency offset. The FFT-based algorithm is suitable for all modulation types and the latter for PAK and pulse-amplitude modulation. Compared with the FFT-based algorithm, the Luise algorithm requires fewer hardware resources. In the Luise algorithm, there are $M$ register weighted by $1/(N - i)$, $i = 1, 2, \ldots, M$, where $N$ is the number of samples. In this paper, $N$ is set to be 512 to facilitate the bit shift operation and $M$ is chosen as 3. Three coefficients ($1/511$, $1/510$, and $1/509$) are approximated as $1/512$. Thus, the corresponding three multiplications can be replaced with the only one multiplication with $1/512$ which can be implemented with a 9-bit shift operation.

IV. TRANSMITTER MODEL DESIGN

The SDR structure of the transmitter system model is shown in Fig. 3, in which an FPGA SoC (Zynq 7020) is deployed as the baseband processor to generate baseband signals and an RF transceiver (AD-FCOMMS3-EBZ) is used as the front end to transmit an RF waveform.

As shown in Fig. 3, the transmitter model mainly consists of an onboard computer/PC, an LDPC encoder, an OQPSK symbol mapping block, a pulse shaping filter, a controller, and an RF transceiver. Among these baseband blocks, the LDPC
encoder and the OQPSK symbol mapping are implemented with the “black box” tools in the HDL Coder [28], while other models are implemented by building Simulink blocks. Black box provides an efficient way to embed the customized HDL code, such as Verilog, in the Simulink environment. All these modules are designed and optimized in the HDL Coder for targeting to an FPGA [29].

Data bits are generated by the onboard computer/PC and then sent to the FPGA through the Ethernet interface. The input signal to the FPGA is a 32-bit bus, and only the data bits are extracted to go through the LDPC encoder. Subsequently, the coded bits are mapped into OQPSK symbols and then go through a square root raised cosine filter. The pulse shaping signals are upconverted to an RF by a carrier wave at 2.4 GHz through the AD-FCOMMS3-EBZ radio board. Finally, the RF signals are transmitted through the antenna.

A. LDPC Encoder

The $H$ matrix of EG-LDPC (255, 175) can be generated by taking the incidence vector of a line in EG $(2, 2^4)$ and cyclically shifting the incidence vector [20]. Such a construction method not only generates a code with a large minimum distance (distance of 17) but also boosts the convergence of the decoding process. The large minimum distance helps the decoder to lower the error floor and improve the BER performance. Since the proposed LDPC code converges quickly, only a small number of decoding iterations are needed to reach the required BER performance. This will be demonstrated in Section V-B. The $H$ matrix of this code is a square matrix of size $255 \times 255$ with both row and column degrees of 16.

Fig. 4 shows the structure of the LDPC encoder, which is comprised of a finite-state machine (FSM) controller, a bit buffer, information bits selector and a parity check bit generator. This encoder is implemented based on the matrix $H_j$. When a new frame comes, both the pilot bits and information bits directly go through the encoder. At the same time, the information bits are stored in the bits buffer. When the parity check bit $p_j$ is calculated, the related information bits $\{d_i : i \in B_j\}$ are selected in the information bits selector. Simultaneously, these selected bits and the parity check bit $p_{j-1}$ are fed into the parity check bit generator. The parity check bit generator is a module that has $d_{\text{max}} + 1$ ports and returns the XOR value of all the input bits. If the number of the selected bits is less than $d_{\text{max}}$, values on the other port will be set to 0. Since the oversample factor is configured as four, there are four clock cycles in each symbol. A half rate frequency divider is needed and configured as the input clock of this encoder so that only two bits are generated in one symbol period. The LDPC encoder outputs 2 bits per symbol period that are used as the input of OQPSK modulator.

It can be found that there is only one group of XOR operations required in this encoder and this group of XOR is shared for 80 times. The information bits selector, also an FSM controller, determines which information bits are selected in the corresponding clock period. To verify how many resources the encoder described in Algorithm 1 can save, hardware resources of encoders designed in the two algorithms are compared. The resource utilization of the two encoders is summarized in Table I. The comparison indicates that Algorithm 2 helps to save hardware resources significantly. Although Algorithm 1 requires much more resources, it can be adopted in the scenario that needs high throughout since the parallel use of $(n-k)$ groups of XOR enables $(n-k)$ parity check bits to be generated simultaneously.

V. RECEIVER MODEL DESIGN

This section will describe the receiver system model and hardware considerations of designing a receiving link. In the receiver, an RF transceiver SoC (AD-FCOMMS3-EBZ RF transceiver) is deployed as the front end to receive RF signals and an FPGA SoC as the baseband processor to process the baseband signals. The SDR structure of the receiver model is shown in Fig. 5. As shown in Fig. 5, RF signals are downconverted to baseband signals in the RF transceiver and then fed into the FPGA under the controllers’ control. The baseband processor is mainly consisted of seven components: AGC, matched filter, frequency compensation, timing recovery, frame synchronization, soft demodulation, and LDPC decoding. Among these modules, soft demodulation and LDPC decoding blocks are implemented using HDL toolbox. A general description of these blocks is given as follows.

The AGC block is designed to regulate the received signal strength and keep the receiver working within the normal operating range aiming at overcoming the potential signal fading and sudden increase in signals’ power [30]. The AGC algorithm boosts the gain if the received signal has weak amplitude, whereas the gain is attenuated to avoid signal
clipping and nonlinear degradation if the received signal has strong amplitude. With the AGC algorithm, the average peak amplitude is dynamically adjusted to a stable value, which is necessary for the subsequent modules.

The frequency estimation comprises two stages: CFC and FFC. The CFC performs a rough order estimate on the frequency error between the expected frequency and the observed frequency from the received signals. The FFC provides a more precise estimation of the remaining small frequency error.

The header of the frame is found in the frame synchronization block so that the payload symbols can be located. Log-likelihood ratio (LLR) values are calculated in a soft-decision modulator. Once the decoded bits and valid signal are available, they are packed into a pair and sent out. After the decoded bits and valid signal are separated in the host computer, the decoded bits are converted into readable message and printed on the screen.

A. Simplification of Magnitude Detection in AGC

AGC is typically designed with a closed-loop feedback structure that mainly includes a decision feedback equalizer (DFE) and a magnitude detection block. In this AGC module, the loop gain and the reference value are set to 0.1 and 0.5, respectively. The DFE generates a feedback value according to the magnitude of the input signal, loop gain, and reference value. This feedback value is dynamically calculated and acts as an amplifier or attenuator. The detection of signal’s magnitude is involved in the square and root square calculations. Conventional methods to detect the signal include the adoption of IP core or CORDIC algorithms. In this paper, an improved scheme to detect signal’s magnitude is proposed. The implementation structure is shown in Fig. 6. The magnitude detection block gives an approximate value of the amplitude by implementing the following operation [31]:

\[
\sqrt{I^2 + Q^2} \approx |L| + \lambda \ast |S|
\]  

(3)

where \(L = \max(I, Q)\), \(S = \min(I, Q)\), \(I\) and \(Q\) are the amplitudes of in-phase and quadrature components, respectively, and \(\lambda\) is a positive number between 0 and 1. The approximation suggests that the calculation of a signal’s amplitude can be realized by the linear combination of amplitudes of in-phase and quadrature components.

We ought to find an optimal \(\lambda\) value that can minimize the error caused by the approximation. \(\lambda\) is enumerated with a step size of \(1/27\) to find its optimal value. For each \(\lambda\), 10000 I and Q samples are random values within the range \([-10, 10]\). The actual amplitude \((I^2 + Q^2)^{1/2}\) and the approximate amplitude
\[ |L| + \lambda |S| \] of these samples are calculated. The mean square error (mse) of the approximation can also be obtained. Fig. 7 shows the mse at difference \( \lambda \), which indicates that the optimal \( \lambda \) value equals 0.3125.

### B. Simplified LDPC Decoder

As described in Section IV-A, the \( H \) matrix of EG-LDPC (255, 175) is of size 255 \( \times \) 255. A total of 255 CNPs, 255 VNPs, and 8160 (16 \( \times \) 255 \( \times \) 2) register units are needed in order to design a decoder with a fully parallel architecture. A decoder with 255 CNPs may bring 0.4-dB gain at the BER of 10\(^{-7}\) compared with the one with 80 CNPs; however, it requires two times more hardware resources. In this paper, a simplified LDPC coder based on the matrix \( H' \) is designed to save hardware resources. The new matrix \( H' \) can be represented by a Tanner graph [26]. There are 80 CNs and 255 VNPs in the graph, which correspond with 80 CNPs and 255 VNPs. Each row of \( H' \) has a regular row degree of 16; however, each column has an irregular column degree. Suppose that the \( j \)th column of \( H' \) has a degree of \( d_j \). Accordingly, each CN is connected to 16 VNPs and the \( j \)th VN is connected to \( d_j \) CNs.

The structure of the simplified LPDC decoder is shown in Fig. 8, which includes an FSM controller, LLR buffer, channel information LLR register, vtc registers, ctv registers, 80 CNPs, 255 VNPs, and bits buffer. vtc registers store the message that VNPs updated and ctv registers store the message that CNPs updated. Since the new matrix \( H' \) has 80 rows and each row has a regular degree of 16, there are only 1280 (16 \( \times \) 80) units needed in the ctv and vtc registers, respectively. The decoding processes can be described as four steps: buffer the LLR values, CNs and VNPs alternatively update, make hard decision, and send the decoded bits out.

First, an LLR start signal is activated when LLR values are received from the modulator. Once the LLR valid signal is high, every two LLR values are stored into the LLR buffers in each symbol period. The FSM controller distributes each pair of LLR values to the corresponding buffer. It takes 128 symbol periods to store a frame of LLR values. Once a whole frame of LLR values are buffered, the values in the LLR buffer are moved into the channel information LLR register. Then, CNPs and VNPs begin to work. Second, 80 CNPs simultaneously read data from the vtc registers and then update values and store the updated values into the ctv registers, which take two symbol periods. After CNPs’ operations, 255 VNPs work in a similar way: read data from the ctv registers and then update values and store the updated values into the vtc registers, which take two symbol periods as well. A total of four symbol periods are needed to finish one iteration. Usually, more iterations may result in a better performance but also lower the throughput. To make a compromise, CNPs and VNPs alternatively perform read-and-update operations for 20 times. Third, when VNPs finish the 20th iteration, hard decision is made based on the summation of channel information and the updated values from all the CNPs. Then, VNPs send the whole frame of bits to the bits buffer and simultaneously the FSM controller generates a valid signal, which indicates that decoded bits are available. Finally, every two decoded bits are sent out from the bits buffer in each symbol period.

During the CNs’ update processes, there are 80 CNPs working simultaneously. Each CNP reads 16 values from the vtc registers sequentially. Here, the corresponding 16 positions are only decided by the positions of “1” in the row of \( H' \) [32]. In hardware implementation, the main operation of CNPs’ update is to find the minimum and second minimum absolute values among these 16 values. An efficient way to find these two values is implementing a three-level comparator tree, as shown in Fig. 9 [33], [34]. There are seven same units duplicated, and each one is used to find the minimum and subminimum values from the four inputs. The comparator tree is able to find these values within one clock cycle. When the minimum and the subminimum are found, both are scaled by 0.75 [26].

Similarly, there are 255 VNPs working simultaneously during the VNPs’ update processes. The \( j \)th VNP only reads \( d_j \) values from the corresponding \( d_j \) positions of the vtc registers. Here, the corresponding \( d_j \) positions are only decided by the positions of “1” in the \( j \)th column of \( H' \) [32]. The values to be updated are simply the summation of the channel information and the \( (d_j - 1) \) values that are selected from \( d_j \) values read by \( j \)th VNP [26]. More details of CNs’ and VNPs’ update processes can be found in [26].

The modified min-sum decoder is implemented with a fixed point data representation. The reduction of bit width may lead to reduction of area, power consumption, and hardware resources. However, this may also bring degradation of the decoding performance. It is necessary to find an efficient fixed point scheme that uses bit width as small as possible and keeps an acceptable performance. The BER versus \( E_b/N_0 \) for different simulation schemes is shown in Fig. 10; 300-bit errors are collected in each \( E_b/N_0 \). The dashed lines show the simulation results in a different fixed point precision, and the solid lines show the float precision performances for different numbers of iterations. The solid line with pentagram shows the performance of the decoder that is designed with 255 CNPs and 255 VNPs. Let \((i, f)\) denote the fixed point simulation scheme with the total of \((1 + i + f)\) bits, where 1 bit is used...
CAI et al.: LOW-POWER SDR DESIGN ON AN FPGA FOR INTERSATELLITE COMMUNICATIONS

Fig. 8. Simplified LDPC decoder with a fully parallel structure.

Fig. 9. Structure of the comparator to find the minimum and second minimum values.

Fig. 10. Simulation results of the modified min-sum decoder.

for sign bit, $i$ bits for integer part, and $f$ bits for fractional part of the LLR value. The simulation results show that this code using 7 bits (1 bit for sign, 4 bits for the integer part, and 2 bits for fractional part) or 6 bits (1 bit for sign, 4 bits for the integer part, and 1 bits for fractional part) representation easily suffer from a higher error floor. Considering the hardware resources and decoding performance, $(5, 1)$ is a good choice to represent this code. It is shown that decoding with 50 iterations only improves the performance by 0.1 dB at a BER of $10^{-6}$ than the one with 10 iterations and almost has the same performance with the one in 20 iterations. It is also demonstrated that the performance of this code under a modified decoder is significantly improved at $E_b/N_0$ higher than 4.5 dB compared with the rate $1/2$ LDPC code from [9].

VI. IMPLEMENTATION AND RESULTS

To validate the proposed design, over-the-air transmissions experiments are conducted on the SDR platform. The test procedure can be described as follows. In the transmitter, both pilot and information bits are generated in the host computer, and every 2 bits forms a symbol in the in-phase/quadrature form. These symbols in complex number form are then oversampled to align with the clock rate on the FPGA and fed into the transmitter module on the FPGA. The output from an FPGA is fed to the AD9361 radio that generates the RF waveform and transmits through the antenna. In the receiver, signals are received through the radio and get processed by the receiver module on the FPGA, as described in Section V. Decoded bits are then sent to the host through the output data bus. Finally, the host computer buffers the decoded bits and valid bit and converts the bits to ASCII form for message display.

The frame structure is shown in Fig. 11. A 13-bit Barker code is upsampled by 2 for the QPSK modulation without losing peak autocorrelation characteristic. For easy demonstration, the ASCII representation of “this is the test frame ##” is...
chosen as the 175 information bits, where ## are 2 digits from 00 to 99. The payload is the corresponding LDPC codeword of information data with an extra bit 0 appended. There are 141 symbols (13 for the header and 128 for the payload) in each frame.

A. Test Platform

This section describes how to integrate and run the communication system model. The Workflow Advisor in Simulink [29] is used to generate the HDL code for all blocks in the transmitter and receiver except those designed with the black box tools, such as LDPC encoder and decoder. The modules designed with block tools are replaced by the customized HDL code. The Workflow Advisor integrates all the modules to build an entire transmitter and receiver model and generates an FPGA bit file. By downloading the FPGA bit files into FPGAs, all the modules described in Sections IV and V can be targeted to the SDR hardware. The SDR platform is shown in Fig. 12, in which the left is the ZC706 board as the receiver and the right is the ZedBoard as the transmitter. Both FPGA boards hold one antenna and one daughter board (AD-FCOMMS3-EBZ RF transceiver) with the center frequency at 2.4 GHz (adjustable from 70 MHz to 6.0 GHz) and baseband sample rate at 245.76 kHz (adjustable from 200 kHz to 56 MHz). The data rate of our experiments is 122.88 Kb/s. Each FPGA board is connected with the host computer through an Ethernet cable. The transmitter antenna is separated from the receiver antenna by 2 m during the test. When the decoded message is obtained, it is printed on the screen. Fig. 13 shows the received messages during real-time transmission. The accumulative BER value is printed every 50 frames although the actual $E_b/N_0$ is not estimated.

B. Hardware Resources Utilization and Power Consumption

The details of hardware resource utilized in the transmitter and receiver are given in Table II. The table shows that the receiver requires significantly more hardware than the transmitter; 71.8% of LUTs are used in the receiver. As analyzed in Section V-B, the decoder if implemented with the $255 \times 255$ H matrix would cost twice more hardware resources. This ZC706 FPGA platform may not be able to fit the entire receiver design if the LDPC decoder is not simplified. An FPGA power estimation shows that the transmitter and receiver require a power consumption of 2.1 and 3.2 W, respectively. As mentioned in Section II, a small satellite usually has a rechargeable battery in the capacity of about 192~400 Wh. The total power consumption of the FPGA-based SDR is about 5 W, which is within the power limitation of an ISC system.

C. Signal Quality in Real-Time Transmission

In digital communications, an error vector magnitude (EVM) value is a comprehensive measurement of the quality of the receiver. The EVM value is calculated as in 4, where $I_j$ and $Q_j$ are the in-phase and quadrature components of the $j$th receive signal, $I_j^-$ and $Q_j^-$ are the in-phase and quadrature components of the $j$th ideal reference signal, respectively, and $N$ is the frame length. In this paper, the ideal reference signals are the corresponding signals on the constellation points

$$\text{EVM} = \sqrt{\frac{1}{N} \sum_{j=1}^{N} \left[ (I_j - I_j^-)^2 + (Q_j - Q_j^-)^2 \right]} \left/ \sqrt{\frac{1}{N} \sum_{j=1}^{N} \left[ (I_j^-)^2 + (Q_j^-)^2 \right]} \right..$$
When the transmission tests are conducted, the signals out of the frame synchronization module are collected and the BER values for the collected signals are obtained. According to the collected signals and the reference signals, the EVM values can be calculated. From the theoretical BER versus $E_b/N_0$ curve, $E_b/N_0$ can be estimated. Table III shows the calculated EVM values and the estimated $E_b/N_0$ values in different BER levels. It is shown that the calculated EVM values match the theoretical ones [35]. A smaller EVM indicates that the collected signals are closer to the ideal signals.

**D. Comparison With Other Works**

Reed–Solomon (RS) codes are widely used in consultative committee for space data system’s (CCSDS’s) space communications protocol specifications. Based on NASA’s research on RS codes, the CCSDS has specified RS (255, 223) as the FEC standard for the telemetry channel coding and RS (255, 239) as the forward and reverse link for an advanced orbiting system. However, the decoding process of these two codes converges very slowly. A comparison with these two codes and the Harris’s convolution code is made in Fig. 14. The RS codes suffer from very poor performance at $E_b/N_0$ lower than 6 dB. Compared with Harris’ half rate convolution code [8], our proposed LDPC decoder in 20 iterations has a 1.6-dB performance gain at the BER of $10^{-6}$. Although Harris’ convolution code shows a better performance than the RS codes at $E_b/N_0$ lower than 6 dB, the convergence is still slower than this LPDC code. The comparison reveals that the simplified LPDC decoder still keeps the property of fast convergence and good performance.

**VII. Conclusion**

The optimized transmitter and receiver models for ISCs are presented and implemented on the FPGA-based SDR platform. The proposed SDR is full-featured and suitable for the power-limited small satellites system. The AGC, frequency compensation, and decoder modules are optimized in order to reduce hardware resource utilization. A simplified decoder for the EG-LDPC (255, 175) code is designed based on the modified MSA. The simplification reduces the hardware utilization by two thirds with only little performance degradation. Furthermore, the transmitter and the receiver are targeted on the FPGA-based SDR platform with a successful demonstration of over-the-air transmission in the S-band.

**REFERENCES**


Xin Cai is currently working toward the Ph.D. degree at Xidian University, Xi’an, China. He was a Visiting Student with the Worcester Polytechnic Institute, Worcester, MA, USA, from 2015 to 2017. His current research interests include information theory, channel coding, and software-defined radio.

Mingda Zhou received the B.S degree from Xi’an Jiaotong University, Xi’an, China, in 2013, and the M.S. degree from the Worcester Polytechnic Institute, Worcester, MA, USA, in 2015, where he is currently working toward the Ph.D. degree. His current research interests include multiple-input multiple-output systems, synchronization, and field-programmable gate array implementation.

Tian Xia (M’01–SM’06) received the B.S. degree in electronics and information engineering from the Huazhong University of Science and Technology, Wuhan, China, the M.S. degree in electrical engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, and the Ph.D. degree in electrical and computer engineering from The University of Rhode Island, Kingston, RI, USA. He is currently a Full Professor at the Department of Electrical and Biomedical Engineering, University of Vermont, Burlington, VT, USA. His current research interests include mixed signal VLSI circuit design and test, and microwave and radio-frequency sensing circuit and system.

Wai H. Fong received the B.S. and M.S. degrees in electrical engineering from George Washington University, Washington, DC, USA, in 1984 and 1992, respectively, and the Ph.D. degree in electrical engineering from George Mason University, Fairfax, VA, USA, in 2015. He is currently with the Goddard Space Flight Center, NASA, Greenbelt, MD, USA. His expertise is in the areas of digital communications specializing in error correction coding and estimation/detection theory. His current research interests include watercolor and oil painting, woodworking, and cosmology.

Wing-Tsz Lee received the B.S degree in electrical engineering from the University of Maryland, College Park, MD, USA, in 2005, and the M.S. degree in electrical engineering from George Mason University, Fairfax, VA, USA, in 2015. She is currently a Communications Engineer at the Goddard Space Flight Center, NASA, Greenbelt, MD, USA. Her current research interests include the area of digital communications.

Xinning Huang (M’01–SM’09) received the Ph.D. degree in electrical engineering from Virginia Tech, Blacksburg, VA, USA, in 2001. He was a Member of the Technical Staff at the Wireless Advanced Technology Laboratory, Bell Labs of Lucent Technologies, Whippany, NJ, USA, from 2001 to 2003. He is currently a Professor at the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA, USA. His current research interests include the areas of circuits and systems, with an emphasis on wireless communications, machine learning, and information security, computer vision, and machine learning.