Hardware Implementation of Space Shift Keying on a Xilinx Zynq Platform

Omar Hiari(✉), Faris Shahin, Samer Alshaer, and Raed Mesleh

German Jordanian University, Amman, Jordan
{omar.hiari,f.shahin,s.alshaer,raed.mesleh}@gju.edu.jo

Abstract. The recent definition of hardware system models for space modulation techniques has provided a pathway to physical implementation of such systems. Space Shift Keying (SSK), being at the forefront of all these definitions, implements a pure form of space modulation that does not require a traditional RF chain that generates baseband symbols. On the other hand, compact, SDR-enabled, platforms powered by computationally powerful SoCs, are also becoming increasingly popular in prototyping wireless systems. In this work, we leverage commercially available SDR-enabled platforms, based on the Xilinx Zynq SoC and the Analog Devices AD9361 analog front end, to implement an entry level SSK system.

Keywords: SDR · Space modulation · Zynq

1 Introduction

Trending wireless protocols such as 5G, have introduced an ever increasing demand on wireless systems to increase data rates, spectral efficiency, and reduce power consumption. One of the approaches to address the aforementioned demands was by introducing Multiple Input Multiple Output (MIMO) systems. Through different implementations, the main idea for MIMO systems is to deploy multiple antennas at the transmitter and receiver sides to exploit multipath propagation properties of a channel. Out of the various MIMO techniques, Space Modulation (SM) has emerged as one of the techniques that could address the demands for future systems [9]. Contrary to traditional implementations of MIMO (Ex. Spatial Multiplexing), SM only requires a single RF, or in some cases zero, chain(s). SM achieves that by transmitting a portion of the information bits through the transmit antenna indexes.

As with all new wireless implementations, there still exists a gap in SM between the theoretical and physical (hardware) implementations. As recently introduced in literature, SM is achievable through a family of eight techniques; Space Shift Keying (SSK), Spatial Modulation (SM), Quadrature Space Shift

Supported by the Deanship of Scientific Research at the German Jordanian University under research grant number SIC 41/2016.

V. Sucasas et al. (Eds.): BROADNETS 2018, LNICST 263, pp. 267–275, 2019.
https://doi.org/10.1007/978-3-030-05195-2_26

Keying (QSSK), Quadrature Spatial Modulation (QSM), Generalized Space Shift Keying (GSSK), Generalized Spatial Modulation (GSM), Generalized Quadrature Space Shift Keying (GQSSK), and Generalized Quadrature Spatial Modulation (GQSM) [6,10,11]. Each implementation offering a different complexity, cost, spectral efficiency, and power consumption depending on application demands.

The emergence of heterogeneous computing platforms, such as the Xilinx Zynq, have provided the ability to accelerate the hardware implementation of theoretical wireless systems [3]. These platforms have been used extensively in other works to achieve Software Defined Radio (SDR) implementations as well as proving out wireless models [5,8,14,15]. Being software defined, these platform provide a lot of flexibility to build on and expand to other implementations.

In this work, we provide the details of physically implementing one of the SM family of techniques, namely SSK, utilizing a Zynq platform. SSK was chosen such that it is a foundational technique that all other implementations would need to build on. The rest of the paper is organized as follows, Sect. 2 introduces the SSK system model, Sect. 3 describes the hardware architecture adopted and the implementation of the system, Sect. 4 describe the results obtained, and finally Sect. 5 concludes this paper.

2 System Model

Figure 1 shows the SSK system model definition. At the transmitter side, incoming serial bits are parallelized and connected to the select lines of an RF switch. Through the select lines states, the RF switch controls which antenna is transmitting at each time instance. The RF input of the switch is merely a periodic carrier signal that could be easily generated in various manners. As a result, the spectral efficiency η of the system therefore is defined merely by the number of antennas N_t deployed in an SSK system and is equal to:

$$\eta_{SSK} = \log_2(N_t) \tag{1}$$

The generated RF signal from the transmitter is transmitted over the MIMO channel matrix $\mathbf{H}$. The transmitted signal received by N_r receive antennas in The N_r-dimensional received vector $\mathbf{y}$ is then given by:

$$\mathbf{y} = \sqrt{E_s}\mathbf{H}\mathbf{x} + \mathbf{n}, \tag{2}$$

where E_s denotes the transmitted energy, $\mathbf{n}$ the noise in the channel, and $\mathbf{H}$ being the complex channel matrix with a dimension of $N_r \times N_t$. At the receiver side, the received signals are then demodulated through an IQ demodulator and processed by an optimum ML decoder to retrieve the source bits. The ML decoder is defined as [7,12,13]:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}_i \in \mathcal{X}} \|\mathbf{y} - \mathbf{H}\mathbf{x}_i\|_{\mathrm{F}}^2, \tag{3}$$

where $\hat{\mathbf{x}}$ denotes the estimated transmitted symbol, $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm, and $\mathbf{x}_i$ is a possible transmitted vector from $\mathcal{X}$, where $\mathcal{X}$ is a set containing all possible transmitted vector combinations.

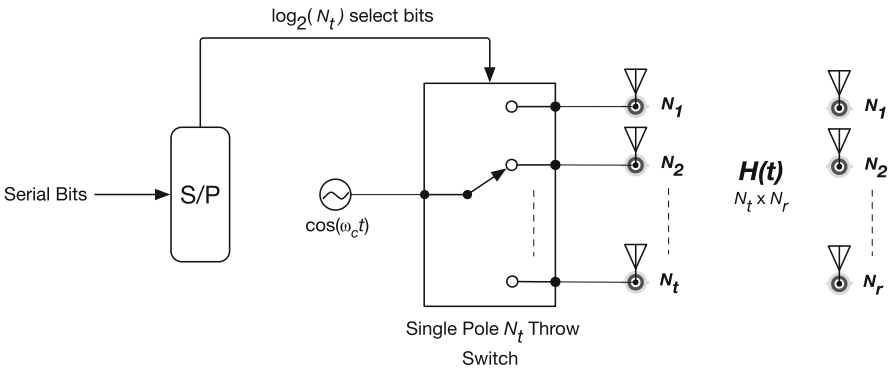

Fig. 1. The SSK system model

3 Hardware System Architecture and Implementation

In order to achieve an SSK realization in hardware, two Xilinx Zynq-based SDR enabled platforms were utilized, one for the transmitter side and another for the receiver [2,3]. The transmitted platform deployed was the Xilinx ZC706 development board with an AD-FMCOMMS5 analog front end attached. The second platform deployed for the receiver was a Zedboard with an AD-FMCOMMS3 analog front end attachment. Both the AD-FMCOMMS3 and AD-FMCOMMS5 incorporate the Analog devices AD9361 [1,4]. The AD-FMCOMMS5 attachment can support four transmit and/or receive chains, on the other hand, the AD-FMCOMMS3 can support only two transmit and/or receive chains. The Zynq device is a heterogeneous System on Chip device that incorporates dual core ARM-A9 cores alongside an FPGA fabric. The platform comes with a base design from Analog Devices providing low level driver support thus providing multiple options to control the low level FPGA and RF front end hardware. The options include; running the hardware with no-OS (i.e. bare metal with low level C drivers), with a Linux image via kernel drivers and GNU radio companion, or a Linux image communicating to MATLAB on a host computer.

The hardware system architecture of the SSK implementation is demonstrated in Fig. 2. On the transmitter side, a Linux image with GNU radio companion is used for creating the transmitter system. Only one of the FMCOMMS5 outputs is activated and is connected to the RF input of an RF switch. The select line of the RF switch is connected to a GPIO output pin coming from the Zynq SoC on the ZC706 board. The RF switch utilized is a SPDT switch with part number ZFSWA2-63DR+ from MiniCircuits, thus allowing for two transmit antenna implementation.

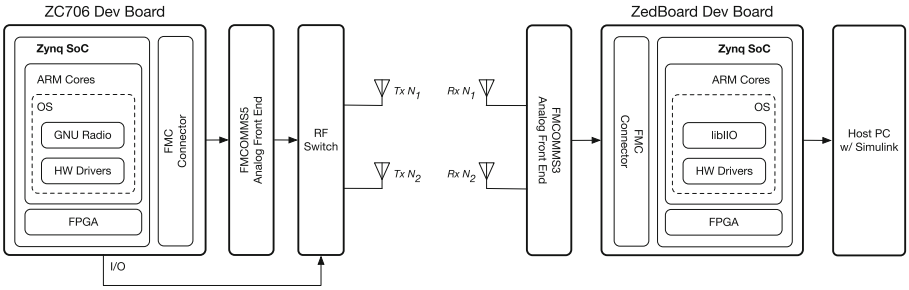

Fig. 2. The hardware system block diagram

The receiver system utilizes both receive chains of the FMCOMMS3 to achieve a 2x2 MIMO system implementation. On the software end, a Linux image with a libIIO object that connects to MATLAB on a PC host is utilized. This allows for simpler post-processing of all the received data.

3.1 Transmitter Implementation

As mentioned earlier, the GNU radio companion was utilized to realize the transmitter implementation. The GNU companion block diagram for the transmitter implementation is shown in Fig. 3. The FMComms5 Sink is a support block provided by Analog Devices to configure and initialize the low level hardware and the AD-FMCOMMS5 RF front-end module. The complex samples required to be transmitted are fed into this sink block. Through the sink block the passband frequency is also determined and it is set to 2.4 GHz. The signal source block generates the desired shape of the transmission signal which the SSK case is a sinusoidal signal. This is the signal utilized to generate the required carrier that connects to the RF switch input.

The TX Frame is a custom block programmed to transmit the required frame. This block contains both bits that are forwarded to the following multiply block and bits that control the switching of the RF switch. The transmit frame consists of 40 bits and is split into three parts as shown in Fig. 4. The first 13 bits contain a barker code sequence to achieve frame synchronization at the receiver side. All of the barker code bits are forwarded to the multiply block to alter the carrier output (enable or disable). During the transmission of the barker code values, the RF switch select line is fixed until the barker code completes transmission. Following the completion of the barker code transmission, a value of 1 is forwarded to the multiply block allowing the carrier output to be enabled for the remainder of the frame transmission. The remainder of the bits in the frame are all transmitted through the antenna indicies by switching the RF switch. Bits 14 and 15 in the transmit frame are pilot bits required for executing channel estimation at the receiver. Finally, the remainder of the frame contains data bits for the message to be transmitted. The switching speed applied to the switch was 20 ms thus achieving a transmission rate of 50 kbps.

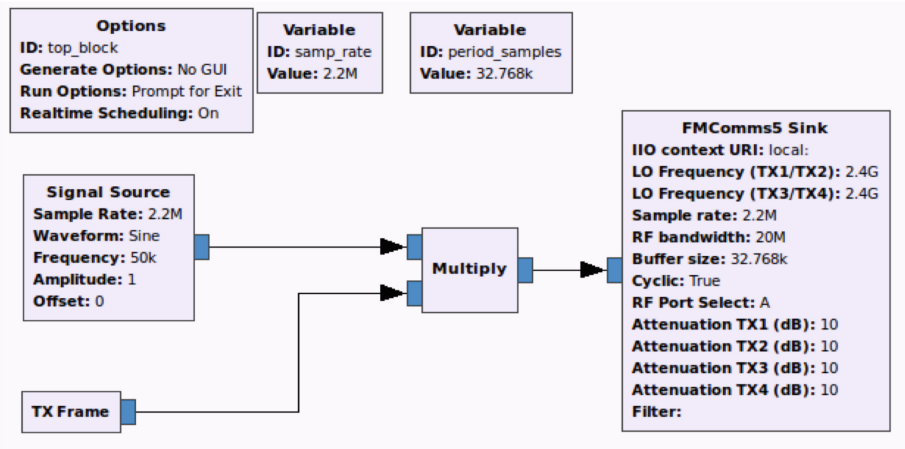

Fig. 3. The GNU radio implementation block diagram at the transmitter

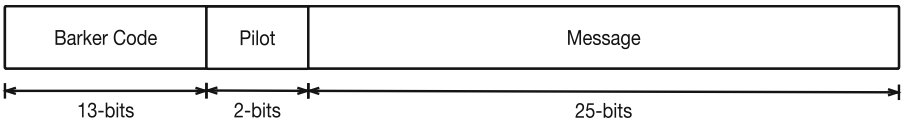

Fig. 4. The transmitted frame

3.2 Receiver Implementation

The receiver system is implemented using Simulink. Simulink provides a hardware interface block that connects and configures the Zedboard platform with the FMCOMMS3. The interface block output is the data received and processed on the low level board hardware. The data format is complex samples that can be further processed in Simulink. The interface block output provides a matrix that combines the samples from the different receive chains.

Figure 5 shows the block diagram implemented in the system on the receiver side. The data received from the board via the interface block is first separated by channel and serialized, such that processing can be performed on each antenna signal separately. Following that, a moving average of the RMS value for the signals is generated to determine the power in the received signals. The signals are then down-sampled for optimized processing. After that the signals are concatenated into a single vector and the frame rebuilt to extract the valid frames through the frame synchronization block. It can be noticed that only one of the channels is connected to the preamble detector for frames. Since frame synchronization is done before any bits are recovered, there isn't a need for information from more than one antenna chain to synchronize frames.

The final block in the receive chain contains the ML decoder and the channel estimator implementations. In this block, the channel estimator leverages the pilot bits and a least squares algorithm in order to determine the channel

parameters. After that is completed, all unwanted data in the frame is stripped away (first 15 bits of barker code and pilot). The generated channel parameters are then fed to the ML decoder as defined in Eq. 3 to recover the transmitted bits. Finally, an error rate block is implemented to determine the BER of the system.

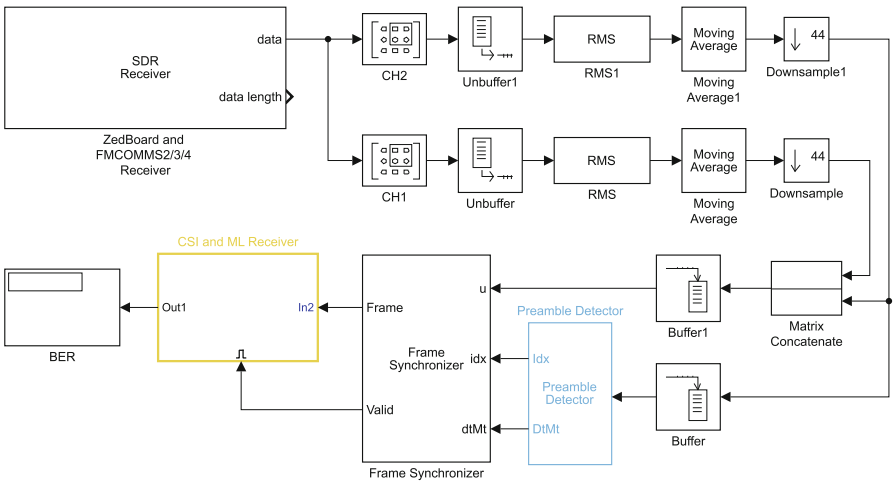

Fig. 5. Simulink model for the SSK receiver

4 Results

A system test run was conducted to determine the correctness of the system implementation. A data file was generated and sent from the transmitter to the receiver in a lab environment. An overall bit error rate of 0.04 was achieved for the transmitted data. Figure 6 shows a plot of the signal received at one antenna at the start of a frame which consists of the 13-bit Barker sequence. For the barker code portion of the transmission, the change in amplitude of the received signal defines whether a zero or a one has been received. Essentially the figure shows that the initial sequence of "1,1,1,1,1,0,0,1,1,0,1,0,1" is being detected.

In Fig. 7, a sample for part of the transmission sequence is shown. The different signal colors indicate the signals on each of the receive channels. The difference in amplitude between the signals is due to the effect of multi-path fading. As a result, comparing the two regions identified on the figure; the switching amplitude levels between signals indicates the antenna that is transmitting. Moreover, it can be noticed in the figure some abnormal transitions appear in between the switching amplitudes. This is due to the effect of switching between antennas on the transmitter side.

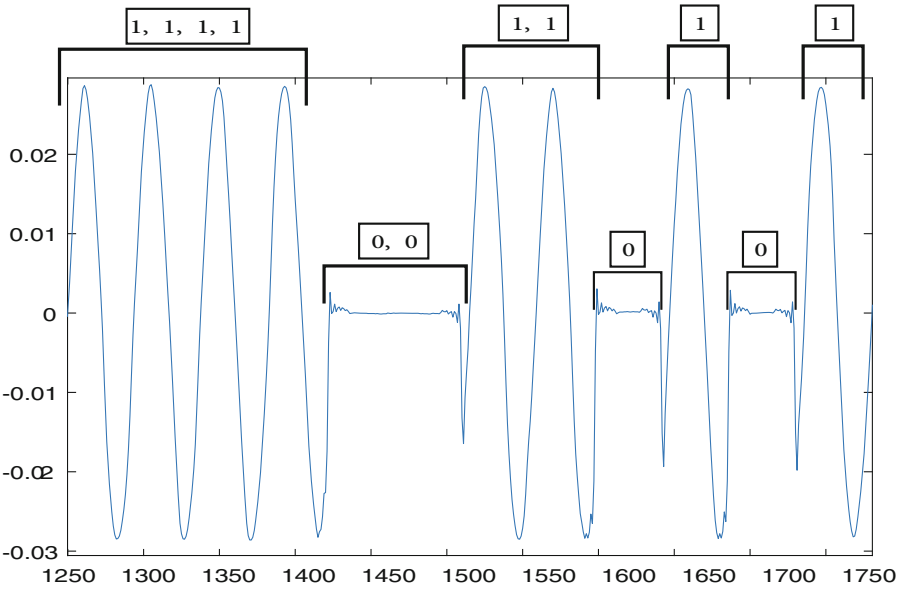

Fig. 6. Barker code signal at the receiver

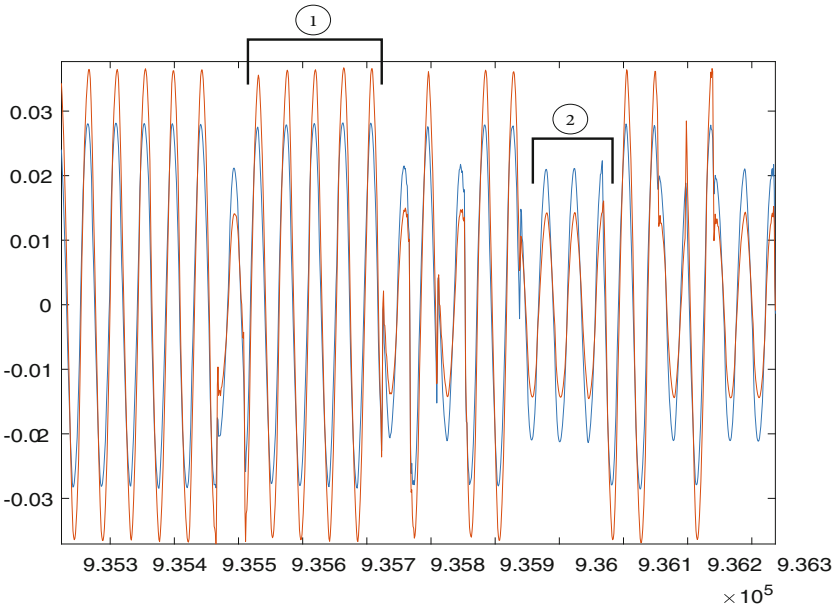

Fig. 7. A sample of the received signal

The results obtained in this work serve as proof of the feasibility of Space Modulation in hardware using an RF switch. However, in the context of future work, this work is still in its preliminary stages as more data, primarily performance data at different antenna configurations and SNRs, needs to be collected. There is also much hardware imperfections impact that need to be studied. Finally, out of the family of space modulation SSK is one of few others presented in literature. This platform would be a good entry level to implement other space modulation techniques.

5 Conclusion

In this work we demonstrate the hardware implementation of one of the Space Modulation Techniques, namely SSK. The implementation is achieved utilizing commercial off the shelf SDR-enabled SoC based platforms. The work shows that the realization of SSK is less complicated than traditional MIMO systems. We also define the architecture of the system, the transmitter, and the receiver implementations. This platform facilitates future research for expanding to other hardware implementations in addition to further studying actual performance of Space Modulation physical implementations.

References

1. AD-FMCOMMS3-EBZ, AD9361 Wideband Software Defined Radio Board. http://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/eval-ad-fmcomms3-ebz.html#eb-overview
2. Z7020: Xilinx Zynq-7000 All Programmable SoC
3. Zynq-7000 All Programmable SoC/AD9361 Software-Defined Radio Evaluation Kit. http://zedboard.org/product/zedboard-sdr-ii-evaluation-kit
4. Analog Devices Inc.: AD9361: RF Agile Transceiver
5. Drozdenko, B., Zimmermann, M., Dao, T., Chowdhury, K., Leeser, M.: Hardware-software codesign of wireless transceivers on zynq heterogeneous systems. IEEE Trans. Emerg. Topics Comput. (2017)
6. Hiari, O., Mesleh, R.: Single RF chain transmitter implementing space modulation (2018)
7. Jeganathan, J., Ghrayeb, A., Szczecinski, L.: Spatial modulation: optimal detection and performance analysis. IEEE Commun. Lett. **12**(8) (2008)
8. Junior, S.B., de Oliveira, V.C., Junior, G.B.: Software defined radio implementation of a QPSK modulator/demodulator in an extensive hardware platform based on FPGAs Xilinx ZYNQ. J. Comput. Sci. **11**(4), 598 (2015). https://doi.org/10.3844/jcssp.2015.598.611
9. Mesleh, R., Alhassi, A.: Space Modulation Techniques (2018). https://www.wiley.com/en-us/Space+Modulation+Techniques-p-9781119375692
10. Mesleh, R., Hiari, O., Younis, A.: Generalized space modulation techniques: Hardware design and considerations. Phys. Commun. **26**, 87–95 (2018)
11. Mesleh, R., Hiari, O., Younis, A., Alouneh, S.: Transmitter design and hardware considerations for different space modulation techniques. IEEE Trans. Wirel. Commun. **16**(11), 7512–7522 (2017)

12. Mesleh, R., Ikki, S.S., Aggoune, H.M.: Quadrature spatial modulation. IEEE Trans. Veh. Technol. **64**(6), 2738–2742 (2015)
13. Paulraj, A., Nabar, R., Gore, D.: Introduction to Space-Time Wireless Communications. Cambridge University Press, Cambridge (2003)
14. Pu, D., Cozma, A., Hill, T.: Four quick steps to production: Using model-based design for software-defined radio. Analog Dialogue **49** (2015)
15. Shreejith, S., Banarjee, B., Vipin, K., Fahmy, S.A.: Dynamic cognitive radios on the Xilinx Zynq Hybrid FPGA. In: Weichold, M., Hamdi, M., Shakir, M.Z., Abdallah, M., Karagiannidis, G.K., Ismail, M. (eds.) CrownCom 2015. LNICST, vol. 156, pp. 427–437. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24540-9_35