

Article

Design of Universal Code Generator for Multi-Constellation Multi-Frequency GNSS Receiver

Xiaohui Ba ^{1,2,3,*} , Taibin Liu ¹ , Wei Jiang ^{1,2,3}, Jian Wang ^{1,2,3}, Baigen Cai ^{1,2,3}, Linguo Chai ^{1,2,3} and Kun Liang ^{1,2,3} 

- ¹ School of Automation and Intelligence, Beijing Jiaotong University, Beijing 100044, China; 23120233@bjtu.edu.cn (T.L.); weijiang@bjtu.edu.cn (W.J.); wangj@bjtu.edu.cn (J.W.); bgcai@bjtu.edu.cn (B.C.); lgchai@bjtu.edu.cn (L.C.); liangk@bjtu.edu.cn (K.L.)
- ² State Key Laboratory of Rail Transit Control and Safety, Beijing Jiaotong University, Beijing 100044, China
- ³ Beijing Engineering Research Center for Electromagnetic Compatibility and Satellite Navigation, Beijing 100044, China
- * Correspondence: baxiaohui@163.com

Abstract: A multi-constellation, multi-frequency Global Navigation Satellite System (GNSS) receiver is capable of simultaneously receiving signals from multiple satellite constellations across various frequency bands. This allows for increased observations, thereby enhancing navigation accuracy, continuity, effectiveness, and reliability. The spread spectrum code structures used in satellite navigation signals differ among systems. Compatible code generators are employed in multi-constellation, multi-frequency GNSS receivers to support tasks such as signal acquisition and tracking. There are three main types of spread spectrum code structures: Linear Feedback Shift Register (LFSR), Legendre sequences, and Memory codes. The Indian Regional Navigation Satellite System (IRNSS) released the L1-SPS (Standard Positioning Service) signal format in August 2023, which utilizes the Interleaved Z4-linear ranging code (IZ4 code) as its spread spectrum code. Currently, there is no universal code generator design compatible with the IZ4 code. In this paper, a proposed universal code generator is based on the hardware structure of the IRNSS IZ4 code generator. It achieves compatibility with all LFSR-based spread spectrum codes and enables parallel generation of multiple sets of GNSS signal spread spectrum codes, thereby improving hardware utilization efficiency. The proposed structure is implemented and validated using FPGA design, and resource consumption is provided as part of the validation results.

Keywords: GNSS; IRNSS IZ4; universal code generator; FPGA



Citation: Ba, X.; Liu, T.; Jiang, W.; Wang, J.; Cai, B.; Chai, L.; Liang, K. Design of Universal Code Generator for Multi-Constellation Multi-Frequency GNSS Receiver. *Electronics* **2024**, *13*, 1244. <https://doi.org/10.3390/electronics13071244>

Academic Editor: Christos J. Bouras

Received: 14 January 2024

Revised: 19 March 2024

Accepted: 22 March 2024

Published: 27 March 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the establishment and advancement of the four major global satellite navigation systems, namely GPS, Galileo, GLONASS, and BDS, alongside regional navigation systems like India's IRNSS and Japan's QZSS, the satellite navigation system has evolved into a multi-constellation navigation ecosystem with over 130 satellites offering navigation services. Each satellite navigation constellation broadcasts multiple civilian signals. For instance, the third-generation global Beidou system transmits various civil signals including B1C, B1I, B2a, B2b, and B3I.

A receiver that supports multiple satellite constellations and multiple frequency bands is capable of receiving signals concurrently from these constellations. This allows for the receiver to acquire additional measurements, enabling multi-mode integrated positioning. Such integration improves the precision, continuity, effectiveness, and reliability of navigation.

The proposed universal spread spectrum code generator can be applied to the design of multi-mode and multi-frequency GNSS receiver chips in order to enhance compatibility and minimize design costs. However, the introduction of IZ4 has increased hard-

ware complexity. Therefore, it is crucial to address the resulting increase in hardware structure complexity.

Significant global research has been conducted on the design of compatible code generators [1–5]. The linear feedback shift register (LFSR) is widely used for generating a majority of GNSS signals. In Ref. [2], a universal ranging code generator is proposed, comprising two 14-bit linear feedback shift registers (SR1 and SR2). These shift registers can be combined to form a 28-bit shift register. Additionally, this paper presents a structural scheme for the GPS L2C ranging code generator using the Fibonacci configuration. By setting different initial values to determine the feedback tap position, an m-sequence structure is achieved. In Ref. [3], a memory-based approach is proposed for generating spread spectrum codes. This approach replaces the previously used dedicated logic modules and instead precomputes and stores the respective spread spectrum codes for each signal in a memory unit. When the receiver receives a signal, it retrieves the corresponding spread spectrum code from memory for processing. The exception to this memory code method is GPS L2CL, which has a code length of 767,250. As a result, the required memory size per channel increases. Each common channel necessitates a minimum of two blocks of 16 kbit RAM and a 27-bit LFSR as essential resources. Ref. [4] presents the design of a spread spectrum code generator architecture that supports multi-constellation and multi-frequency signals. The code generator accommodates three types of spread spectrum codes: linear feedback shift register (LFSR)-based codes, memory codes, and Weil codes. It uses two M-long shift registers to support the generation of spread spectrum codes for satellite signals through time division multiplexing. In Ref. [5], an area-efficient universal code generator based on linear feedback shift registers is proposed. It can be configured with shift registers and tap positions to generate different signal spread spectrum codes, resulting in significant savings of hardware resources. However, its support for signals is not comprehensive, as the code generator cannot handle IRNSS, QZSS L6, and other related signals. Furthermore, the four major navigation systems, namely GPS, Galileo, GLONASS, and BDS, require separate code generator structures.

In August 2023, India publicly released the NavIC Interface Control Documentation (ICD) for L1 SPS [6], which uses the Interleaved Z4-linear (IZ4) ranging code as its spread spectrum code. The IZ4 code comprises two 55-bit shift registers and a 5-bit shift register. Notably, the code generator for IZ4 has a significantly longer shift register length compared to other satellite navigation systems. Currently, there are no studies on the compatibility design of the IZ4 code with other satellite navigation systems. In this paper, a spread spectrum code generation scheme is proposed, incorporating the IZ4 code and based on LFSR structure. The structural characteristics of IZ4 are leveraged to support the parallel generation of spread spectrum codes in multi-channel scenarios.

This paper is structured as follows: the first section introduces the background and research status of compatible code generators, Section 2 discusses spread spectrum code generation schemes for all signals, Section 3 is divided into two parts presenting existing shift register-based schemes and a proposed scheme supporting various shift register structures, Section 4 provides experimental results on the accuracy of generated spread spectrum codes and resource utilization using an FPGA module, and finally, conclusions and future work are presented in the Section 5.

2. Introduction to GNSS Spread Spectrum Codes

2.1. Introduction to Spread Spectrum Codes

There are three primary methods for generating spread spectrum codes in GNSS signals: LFSR-based codes, memory-based codes, and Weil codes. LFSR is the predominant method used for generating spread spectrum codes in most satellite navigation signals [7]. Memory-based codes involve storing pseudo random noise (PRN) codes in memory and retrieving them for use when the receiver receives the signal. This method is employed for generating Galileo E1B/C and E6B/C signals. Currently, Weil codes are exclusively utilized in GNSS systems for generating GPS L1C and BDS B1C signals [8–10].

The majority of signals in GPS, GLONASS, Galileo, and BDS can be generated using LFSR structures [11–14]. Table 1 provides the spread spectrum code generation methods for each signal, including BDS B1I, B2I, B3I, B2a-d, B2a-p, B2b-I, GPS/QZSS L1C/A, L2 CM, L2 CL, L5I, L5Q, Glonass L1OF, L2OF, L1OCd, L1OCp, L2OCp, L3OCd, L3OCp, Galileo E5a-I, E5a-Q, E5b-I, E5b-Q, IRNSS L5-SPS, S-SPS, L1-SPS, QZSS L1S, L5S, L6D, L6E, and SBAS L1, L5 [15]. These codes can be generated by configuring LSFR to produce the corresponding spread spectrum code sequences for each signal.

Table 1. GNSS Spread Spectrum Code Characteristics.

GNSS	Bandwidth	Signal	Code Structure	Code Type	Length	Code Rate	Note	Markings
BDS	B1	B1I	LFSR (11b, 11b)	Gold code	2046	2.046 Mcps	Code generator is the same as B1I The subcode of B2a-p is the Weil code	①
	L1	B1C	Legendre sequence	Weil code	10,230	1.023 Mbps		②
	B2	B2I	LFSR (11b, 11b)	Gold code	2046	2.046 Mcps		③
	L5	B2a (B2a-d/B2a-p)	LFSR (13b, 13b)	Gold code	10,230	10.23 Mcps		④
	B3	B2b-I	LFSR (13b, 13b)	Gold code	10,230	10.23 Mcps		⑤
		B3I	LFSR (13b, 13b)	Gold code	10,230	10.23 Mcps		⑥
GPS/QZSS	L1	L1C/A	LFSR (10b, 10b)	Gold code	1023	1.023 Mcps		⑦
		L1C	Legendre sequence	Weil code	10,230	1.023 Mcps		⑧
		L2CM	LFSR (27b)	Equivalent m-sequence [2]	10,230	0.5115 Mcps		⑨
	L2	L2CL	LFSR (27b)	Equivalent m-sequence [2]	767,250	0.5115 Mcps		⑩
		L5I	LFSR (13b, 13b)	Gold code	10,230	10.23 Mcps		⑪
	L5	L5Q	LFSR (13b, 13b)	Gold code	10,230	10.23 Mcps		⑫
GLONASS	L1	L1OF	LFSR (9b)	M-sequence	511	0.511 Mcps	Same as L1OF	⑬
		L1OCd	LFSR (10b, 10b)	Gold code	1023	0.5115 Mcps		⑭
		L1OCp	LFSR (12b, 6b)	Kasami code	4092	0.5115 Mcps		⑮
	L2	L2OF	LFSR (9b)	M-sequence	511	0.511 Mcps		⑯
		L2OC (L2OCp)	LFSR (14b, 7b)	Kasami code	10,230	0.5115 Mcps		⑰
	L3	L3OC (L3OCd/L3OCp)	LFSR (14b, 7b)	Kasami code	10,230	10.23 Mcps		⑱
Galileo	E1	E1	Memory code	Memory code	4092	1.023 Mcps		⑲
	E5	E5a (E5a-I/E5a-Q)	LFSR (14b, 14b)	Gold code	10,230	10.23 Mcps		⑳
		E5b (E5b-I/E5b-Q)	LFSR (14b, 14b)	Gold code	10,230	10.23 Mcps		㉑
	E6	E6B/C	Memeory code	Memeory code	2046	5.115 Mcps		㉒
IRNSS	L5	L5-SPS	LFSR (10b, 10b)	Gold code	1023	1.023 Mcps	Code generator is the same as GPS L1C/A	㉓
	S	S-SPS	LFSR (10b, 10b)	Gold code	1023	1.023 Mcp	Code generator is the same as GPS L1C/A	㉔
	L1	L1-SPS	LFSR (55b, 55b, 5b)	Interleaved Z4-linear (IZ4) sequences	10,230	1.023 Mcp	Has the longest shift register	㉕
QZSS	L1	L1S	LFSR (10b, 10b)	Gold code	1023	1.023 Mcps	Code generator is the same as GPS L1C/A	㉖
	L5	L5S	LFSR (13b, 13b)	Gold code	10,230	10.23 Mcps	Code generator is the same as GPS L5I and L5Q	㉗
	L6	L6D	LFSR (20b, 10b)	Kasami code	10,230	2.5575 Mcps		㉘
		L6E	LFSR (20b, 10b)	Kasami code	10,230	2.5575 Mcps		㉙
SBAS	L1	L1	LFSR (10b, 10b)	Gold code	1023	1.023 Mcps	Code generator is the same as GPS L1C/A	㉚
	L5	L5	LFSR (13b, 13b)	Gold code	10230	10.23 Mcps	Code generator is the same as GPS L5I and L5Q	㉛

2.2. Spread Spectrum Code Generation Methods Based on LFSR

The LFSR-based spread spectrum code generator primarily comprises a shift register and a feedback tap circuit. Based on the specific configuration of LFSR, the GNSS PRN generation method utilizing LFSR can be categorized into m-sequence, Gold code, Kasami code, and IZ4 code [16].

2.2.1. m-Sequence

The m-sequence is a pseudo-random sequence of length $2^N - 1$ using a register of length N shifted by linear feedback. The m-sequence has good autocorrelation and cross-correlation characteristics.

Signals that utilize m-sequence codes include GPS L2C [2], GLONASS L1OF, and L2OF signals. The structure of the spread spectrum code generated by the m-sequence is typically simpler. For example, the GLONASS L1OF signal [12] employs the simplest PRN code generator structure, as depicted in Figure 1. It consists of a 9-bit shift register with feedback taps at Positions 5 and 9. The register is initialized with a value of one. The generated sequence corresponds to an m-sequence. The period of this sequence is $L = 2^9 - 1 = 511$ chips.

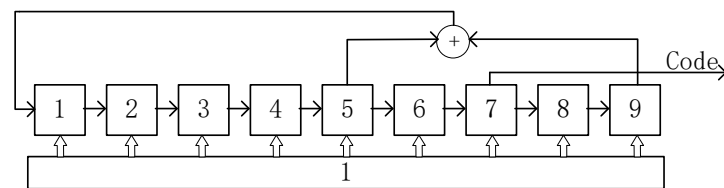


Figure 1. Schematic Diagram of the Glonass L1OF/L2OF Code Generator.

For the GPS L2C signal, its ranging code is a truncated m-sequence. The code generator consists of a 27-bit shift register that generates different pseudocodes for different satellites by configuring different initial phases. It should be noted that the output of the last bit of the code generator is fed back to both the first bit and the intermediate bits. The code generator specified in the ICD [13] uses the Galois configuration (as shown in Figure 2), while most generators use the Fibonacci configuration. Each Fibonacci LFSR has an equivalent Galois LFSR that produces exactly the same output sequence [17,18]. The Fibonacci configuration is achieved by configuring the shift register with the correct initial phase to fix the position of the feedback taps, maintaining consistency with previous m-sequence structures.

To simplify the hardware structure of the code generator, the Fibonacci configuration is used in this paper. To convert to a Fibonacci configuration, the correct initial phase of the m-sequence shift register must be found. Therefore, we used a generator in the form of a 27-bit LFSR in Ref. [2] (Figure 3). The feedback tap for this register is given by the polynomial $f(x) = 1 + x^3 + x^4 + x^5 + x^6 + x^9 + x^{11} + x^{13} + x^{16} + x^{19} + x^{21} + x^{24} + x^{27}$. The output sequence is taken from the last bit of the register.

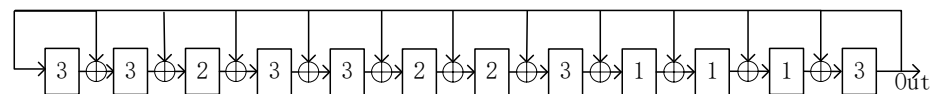


Figure 2. Structure of the GPS L2C Ranging Code Generator in Galois Configuration.

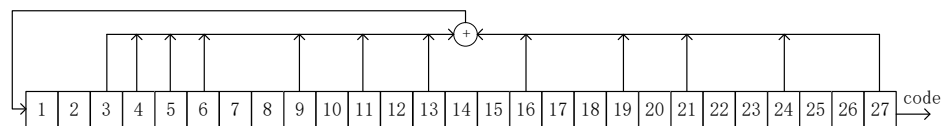


Figure 3. Structure of the GPS L2C Ranging Code Generator in Fibonacci Configuration.

2.2.2. Gold Code

The Gold code is a modulo-2 sum generated by two N -length shift registers with distinct feedback polynomials and initial values. Notably, GPS L1C/A, GPS L5, BDS B1I, GLONASS L1OC, Galileo E5a, E5b, etc., employ the Gold code. In the case of the GPS L1C/A code generator [13], it comprises two registers of identical length (as depicted in Figure 4). The feedback taps are selected from different positions within the two registers. G1 represents the modulo-2 sum of the 3rd and 10th bits, which feeds back into Register 1 with a feedback polynomial of $f(x) = 1 + x^3 + x^{10}$. Similarly, G2 is the modulo-2 sum of

the 2nd, 3rd, 6th, 8th, 9th, and 10th bits, also feeding back into Register 1 with a feedback polynomial of $f(x) = 1 + x^2 + x^3 + x^6 + x^8 + x^9 + x^{10}$. The output sequence is obtained by performing a modulo-2 sum of the sequences stored in Registers G1 and G2.

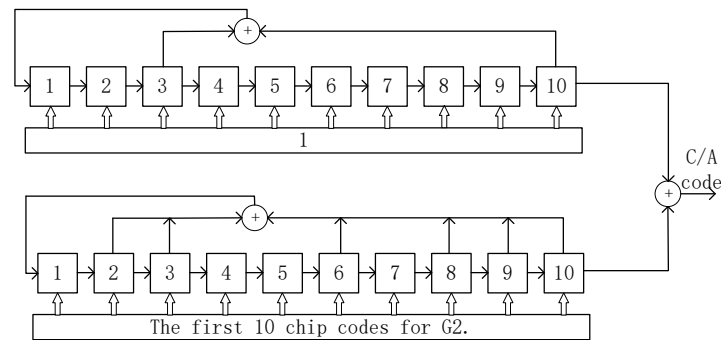


Figure 4. Schematic Diagram of the GPS L1C/A Code Generator.

2.2.3. Kasami Code

Both Kasami and Gold codes are generated as modulo-2 sums of two register sequences. The Kasami code is generated using a shift register of length N and a shift register of length $N/2$, with a maximum length not exceeding $2^N - 1$ [19,20]. The Kasami code is employed by signals such as GLONASS L1OCp, GLONASS L2OCp, GLONASS L3OC, and QZSS L6. In the case of the GLONASS L1OCp code generator [12], it comprises a 12-bit shift register G1, a 6-bit shift register G2, and a feedback tap. The feedback polynomial for G1 is $f(x) = 1 + x^6 + x^8 + x^{11} + x^{12}$ specified as shown in Figure 5, while the feedback polynomial for G2 is $f(x) = 1 + x + x^6$. The initial values of G1 and G2 are stored in the initial registers, and the initial value of G2 varies with the number of satellites. The modulo-2 sum of the outputs from Register 12 of Shift Register G1 and Register 6 of Shift Register G2 yields the GLONASS L1OCp spread spectrum code.

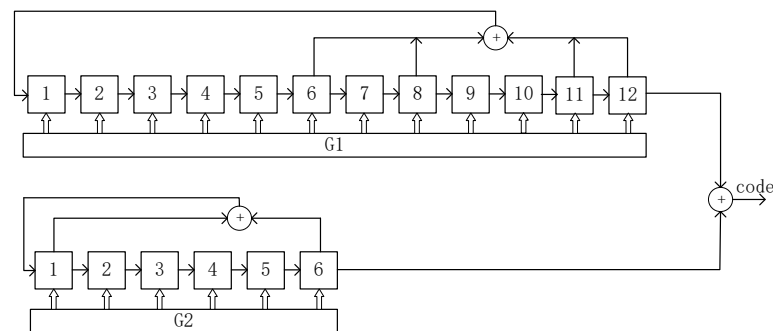


Figure 5. Schematic Diagram of the GLONASS L1OCp Code Generator.

In particular, the QZSS L6D/L6E code generator [15] is introduced, where the structure of L6D is identical to that of L6E except for the initial value of the register. The signal number generator comprises a 20-bit shift register G1 and a 10-bit shift register G2, both with feedback taps. The feedback polynomial for G1 is $f(x) = 1 + x^{14} + x^{16} + x^{19} + x^{20}$ specified as shown in Figure 6, while the feedback polynomial for G2 is $f(x) = 1 + x^3 + x^4 + x^5 + x^6 + x^9 + x^{10}$. The initial value of G1 is stored in the initial register, while the initial value of G2 is fixed at one. By setting and executing the predetermined feedback polynomial and initial value, the modulo-2 sum of the outputs from the 20th bit of Shift Register G1 and the 10th bit of Shift Register G2 yields the spread spectrum code of QZSS L6D/L6E.

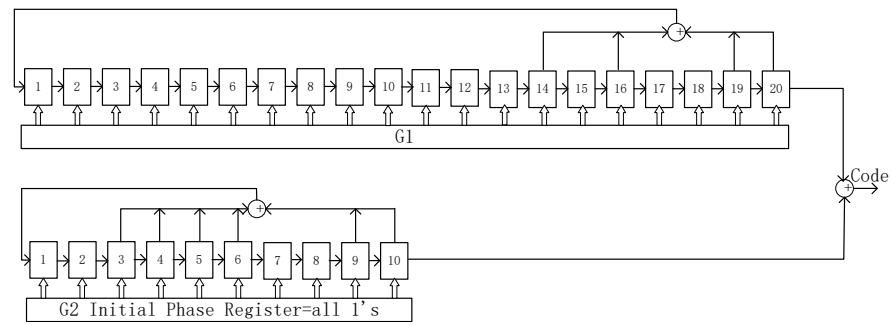


Figure 6. Schematic Diagram of the QZSS L6D/L6E Code Generator.

2.2.4. Interleaved Z4-Linear (IZ4) Ranging Code

In August 2023, India publicly released the Interface Control Document (ICD) for the L1 frequency SPS space signal. As depicted in Figure 7, the code generator structure employed by India is based on the IZ4 scheme. It comprises two 55-bit shift registers, namely R0 and R1, along with a 5-bit shift register denoted as C. The initial value of the register is determined by the input PRN number. The spread spectrum code is generated using the following method [6].

The feedback value of register R0 is determined by the following feedback polynomial:

$$R0(54) = R0(50) \oplus R0(45) \oplus R0(40) \oplus R0(20) \oplus R0(10) \oplus R0(5) \oplus R0(0) \quad (1)$$

In the above equation, the term on the right represents the value of Register R0 at time t , while the term on the left corresponds to the feedback value received by the register at time $(t + 1)$. Initially, at $t = 0$, all registers are initialized, and subsequently, the registers are shifted at each time interval.

For Register R1, the feedback value is obtained by output value R1A of Shift Register R0 and the calculated value R1B of the feedback tap of Register R1. The R0 output, denoted as R1A, is composed of three sub-components: σ_{2A} , σ_{2B} , and σ_{2C} . The resulting modulo-2 sum of R1A and R1B is fed back to the 54th bit of Register R1, denoted as R1(54). The equation below outlines the calculations required to generate the feedback value for Register R1:

$$\sigma_{2A} = [R0(50) \oplus R0(45) \oplus R0(40)] \text{ AND } [R0(20) \oplus R0(10) \oplus R0(5) \oplus R0(0)] \quad (2)$$

$$\sigma_{2B} = ([R0(50) \oplus R0(45)] \text{ AND } R0(40)) \oplus ([R0(20) \oplus R0(10)] \text{ AND } [R0(5) \oplus R0(0)]) \quad (3)$$

$$\sigma_{2C} = [R0(50) \text{ AND } R0(45)] \oplus [R0(20) \text{ AND } R0(10)] \oplus [R0(5) \text{ AND } R0(0)] \quad (4)$$

$$\sigma_2 = \sigma_{2A} \oplus \sigma_{2B} \oplus \sigma_{2C} \quad (5)$$

$$R1A = \sigma_2 \oplus [R0(40) \oplus R0(35) \oplus R0(30) \oplus R0(25) \oplus R0(15) \oplus R0(0)] \quad (6)$$

$$R1B = R1(50) \oplus R1(45) \oplus R1(40) \oplus R1(20) \oplus R1(10) \oplus R1(5) \oplus R1(0) \quad (7)$$

$$R1(54) = R1A \oplus R1B \quad (8)$$

All quantities in Equations (2)–(7) are in-register values corresponding to time t . For Equation (8), the quantity on the left side of the equation represents the value of Register R1 at time $(t + 1)$, the 55th register, i.e., R1(54), which is obtained by modulo-2 summing the values of the right variables R1A and R1B at time t . At time $t = 0$, all registers are initialized and the registers are shifted at each time period.

The IZ4 spread spectrum code is obtained by performing a modulo-2 sum between the first bit of R1, denoted as R1(0), and the first bit of the 5-bit cyclic shift register known as the C register. The C register undergoes a cycle of 10,230 shifts, during which its first bit is directly fed back to the fifth bit. The generator for this process is illustrated in Figure 7.

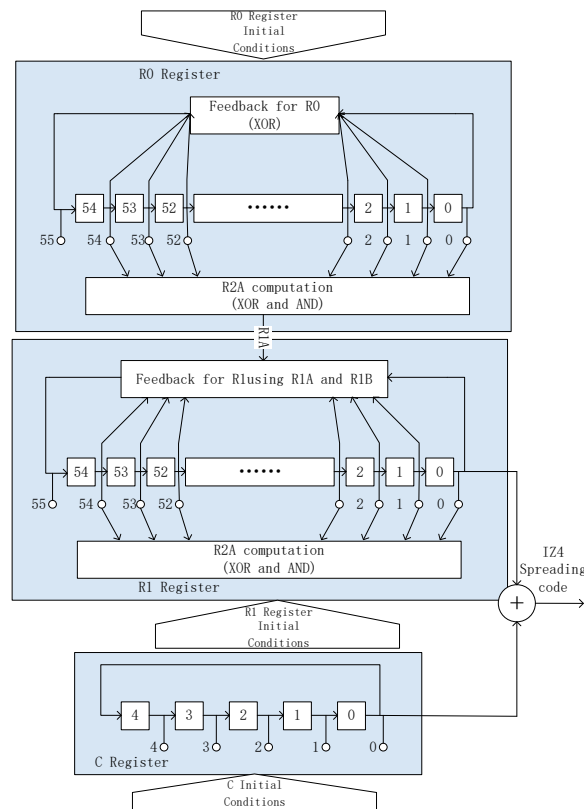


Figure 7. IRNSS IZ4 Code Generator.

The signal spread spectrum code generator is considered as a more complex component within the existing satellite navigation signal generator structure. However, the generation of only the IZ4 signal leads to a relatively lower utilization rate of the hardware structure. Therefore, improvements are required in its structure to enhance the utilization of its hardware resources. Additionally, leveraging this register in combination with other GNSS signals will support the generation of spread spectrum codes for all GNSS signals, further optimizing the hardware utilization.

3. Code Generators

3.1. Universal Code Generators Based on LFSR

Reviewing the relevant literature reveals that currently available universal code generators typically exhibit the following structure:

1. Universal Ranging Code Generator;
2. Area-Efficient Universal Code Generator;

3.1.1. Universal Ranging Code Generator

The Universal Ranging Code Generator, as presented in reference [2] and depicted in Figure 8, comprises two 14-bit linear feedback shift registers, namely SR 1 and SR 2. The initial state of each shift register is stored in their respective CODE STATE 1 and CODE STATE 2 registers. The bitmasks stored in the CODE BITMASK 1 and CODE BITMASK 2 registers control the feedback switches. Similarly, the bitmasks stored in the CODE OUT BITMASK 1 and CODE OUT BITMASK 2 registers determine the output switches.

To maximize the utilization of resources within the universal code generator structure, the structure employs two operational modes.

1. Two separate 14-bit registers (as shown in Figure 8);
2. The two registers are combined to form a 28-bit shift register (as shown in Figure 9).

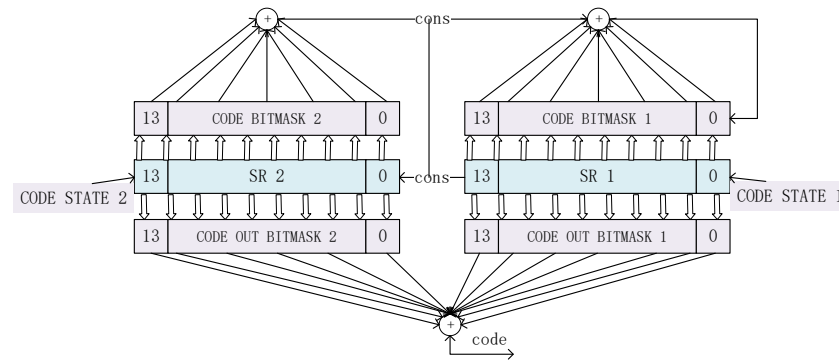


Figure 8. Universal Ranging Code Generator.

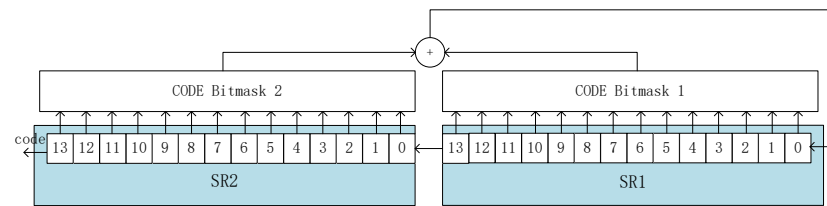


Figure 9. Merged Code Generator.

In the first structure, the PRN code generator comprises two m-sequence generators. Each generator corresponds to a specific PRN number, resulting in distinct initial states and feedback points. During each code cycle, the feedback value is obtained through modulo-2 operations on the bits specified by CODE BITMASK 1 and CODE BITMASK 2 registers. Subsequently, Registers SR 1 and SR 2 undergo left shifts at each cycle, followed by writing the obtained feedback values into the lowest-order bits of SR 1 and SR 2, respectively.

In the second mode, the two shift registers are combined into a single 28-bit shift register, enabling the generation of spread spectrum codes for signals that utilize 27-bit registers (as depicted in Figure 9). During each operational cycle, Shift Registers SR 1 and SR 2 undergo left shifts, with the highest-order bit of SR 1 being written into the lowest-order bit of SR 2. The feedback value is calculated through modulo-2 summation using the bits specified by the CODE BITMASK 1 and CODE BITMASK 2 registers, and the resulting feedback value is written into the lowest-order bit of SR 1.

In each mode, the output sequence is formed by taking the modulo-2 sum of the bits specified by the bit masks stored in the CODE OUT BITMASK 1 and CODE OUT BITMASK 2 registers.

3.1.2. Area-Efficient Universal Code Generator

The conventional universal code generator [5] is characterized by a sophisticated hardware architecture that configures specific initial ROMs, feedback ROMs, and output ROMs for each register bit. By inputting different signal types and satellite numbers, code generators SR1 and SR2 are assigned with their respective initial ROMs, feedback ROMs, and output ROMs. These feedback and output ROMs are utilized by a multiplexer to select different taps for XOR operations, ultimately generating the feedback and spread spectrum codes.

Conventional universal code generators have the capability to generate multiple spread spectrum codes using a single LFSR. However, the configured ROM contains numerous redundant configuration items. The proposed multi-frequency universal code generator in Ref. [5] eliminates unnecessary ROM, MUX, and XOR resources by eliminating unused bits and directly connecting relevant configuration bits. In comparison to conventional multi-frequency universal spread spectrum code generators, this approach significantly reduces redundant hardware resources.

Table 2 presents a summary of the binary ROM values for the spread spectrum code generator, using GPS as an illustrative example. Unused bits are represented as zero,

utilized bits as one, and non-fixed bits as x. Based on the numerical summary, numerous bits are fixed at zero or one, enabling significant reduction in hardware resources. If the shift register corresponds to bit zero, the ROM, MUX, and XOR can be directly removed. On the other hand, if the shift register corresponds to bit one, the ROM and MUX can be eliminated by configuring a direct XOR connection. Therefore, the greater the number of fixed bits at zero and one, the more unnecessary hardware resources can be removed.

Table 2. Fixed Parameters of the GPS Signal Spread Code Generator.

Codes	Feedback ROM1 (13 bit)	Feedback ROM2 (14 bit)	Output ROM1 (13 bit)	Output ROM2 (14 bit)	Initial ROM1 (13 bit)	Initial ROM2 (14 bit)
L1C/A	0001000000100	00001110100110	0001000000000	0000xxxxxxxxxxx	0001111111111	0000111111111
L2CM	1001001010010	01010100111100	1000000000000	0000000000000	xxxxxxxxxxxxx	xxxxxxxxxxxxx
L2CL	1001001010010	01010100111100	1000000000000	0000000000000	xxxxxxxxxxxxx	xxxxxxxxxxxxx
L5 I	1101100000000	01100011101101	1000000000000	0100000000000	1111111111111	0xxxxxxxxxxxx
L5 Q	1101100000000	01100011101101	1000000000000	0100000000000	1111111111111	0xxxxxxxxxxxx
Summary	xx01x0x0x0x0	0xxxxxx1xx1xx	x00x000000000	0x00xxxxxxxxxxx	xxxxxxxxxxxxx	xxxxxxxxxxxxx

As depicted in Figure 10, literature [5] proposes a compatible type code generator structure that enables the generation of spread spectrum codes for GPS, Galileo, GLONASS, and BDS navigation systems. This structure corresponds to four sets of independent code generator structures, one for each system. For example, the code generator designed for the GPS system can generate compatible spread spectrum codes such as GPS L1C/A, L2CM, L2CL, and L5I/Q. Similarly, the code generator tailored for the BDS system can produce compatible spread spectrum codes including BDS B1I, B2I, B3I, B2a, and B2B.

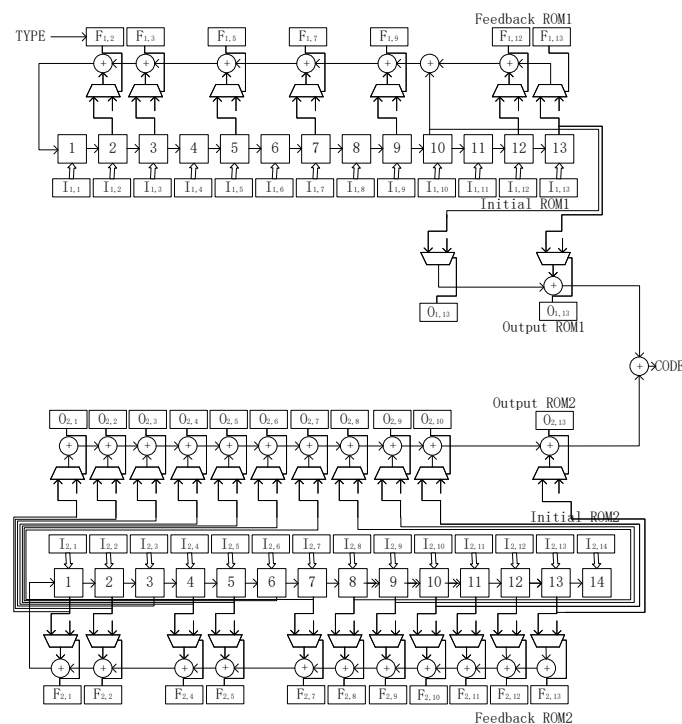


Figure 10. Schematic Diagram of the GPS Universal Code Generator in Ref. [5].

3.2. Proposed Universal Code Generator

In this paper, a register structure is proposed to support the generation of spread spectrum codes based on GPS, BDS, GLONASS, Galileo, QZSS, and IRNSS systems. The register structure is based on the LFSR structure derived from the IZ4 code generator, as depicted in Figure 11. The number of register bits adopts the 55-bit shift register of IZ4, and on this basis, the structure shown in Figure 8 is employed. This allows for the division of registers into different bit lengths, including 6, 7, 9, 10, 11, 12, 13, 14, and 27 bits,

enabling the parallel generation of spread spectrum codes for different signals. For example, simultaneous generation of L5 signal data and the pilot channel spread spectrum code can be achieved, or multiple spread spectrum codes for L1C/A signals and L1C/A and BDS B1I signals can be generated simultaneously by selecting the appropriate control mode. By adopting this approach, not only is the generation of the 55-bit IZ4 spread spectrum code supported, but so is the simultaneous generation of multiple GNSS signal spread spectrum codes, fully utilizing its complex hardware structure.

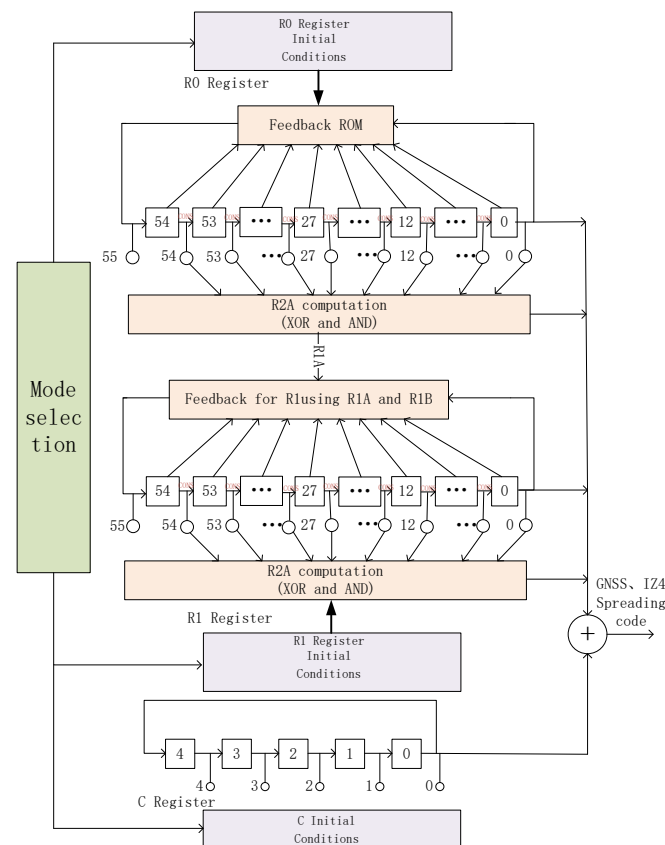


Figure 11. Universal Code Generator.

Table 3 shows the signal generation status according to the selection of the universal code generator mode designed in this paper. The generation of the spread spectrum code is performed by selecting the mode. In mode 0000, the spread spectrum code of the IZ4 data and pilot components can be generated by configuring different initial registers. In mode 0001, the first 0–12 bits of R0 and R1 are configured to generate B2a data components, while bits 13–25 are configured to generate B2a pilot components, and bits 26–38 are configured to generate B2b-I branch components. In mode 0010, bits 0–13 of R0 and R1 are configured to generate E5-I components, and bits 14–27 are configured to generate E5-Q components. In mode 0011, configuring bits 0–12 of register R0 generates the L5 XA sequence, configuring bits 0–12 of R1 generates the L5 XB_I sequence, and configuring bits 13–25 of R1 generates the L5 XB_Q sequence. In mode 0100, the universal code generator simultaneously generates five GPS L1C/A spread spectrum codes (or IRNSS L5-SPS, S-SPS, SBAS L1C/A spread spectrum codes). It configures bits 0–9, 10–19, 20–29, 30–39, and 40–49 in R0 and R1 to generate five groups of L1C/A signals. In mode 0101, the generator simultaneously generates four groups of B3I signals by configuring bits 0–12, 13–25, 26–38, and 39–51 in R0 and R1. If the mode is 0110, it generates four groups of L2C signals using the Fibonacci configuration by configuring bits 0–26 and 27–53 in R0 and R1. In mode 0111, configuring bits 0–9 of R0 and R1 registers generates the secondary code of the IZ4 pilot component. In mode 1000, configuring bits 0–9 and 10–19 of R0 and R1 generates two sets of L1C/A

signals, while configuring bits 20–30 of R0 and R1 generates B1I or B2I signals. In mode 1001, two sets of L1OCd and L1OCp are generated by configuring bits 0–43 of Register R0 and bits 0–31 of R1. In mode 1010, the structure of L2OCp, L3OCd, and L3OCp code generators is completely identical, generating different signal spread spectrum codes by configuring bits 0–41 in R0 and bits 0–20 in R1. In mode 1011, configuring bits 0–13 of R0 and R1 generates E5b-I components, while configuring bits 14–27 of R0 and R1 generates E5b-Q components. In mode 1100, configuring bits 0–39 in Register R0 and bits 0–19 in R1 generates a spread spectrum code for both L6D/E channels.

Table 3. Mode Configuration Table for Universal Code Generators.

Mode	Generate Signal Information	
0000	IZ4 Data and Pilot Component Generation (55 bits)	
0001	B2a Data and Pilot Components (R0 and R1 0–25 bits)	B2b-I branch (bits R0 and R1 26–38)
0010	E5a-I and Q components (R0 and R1 0–27 bits)	
0011	Simultaneous generation of two L5 XA sequences (R0 bits 0–25)	Simultaneous generation of two groups of L5 XB_I and Q (R1 bits 0–51)
0100	Simultaneous generation of five groups of L1C/A (bits R0 and R1 0–49)	
0101	Simultaneous generation of four groups of B3I (R0 and R1 bits 0–51)	
0110	Simultaneous generation of four L2C groups (R0 and R1 bits 0–53)	
0111	IZ4 pilot component secondary code (bits R0 and R1 0–9)	
1000	Two groups L1C/A (R0 and R1 bits 0–19)	B1I or B2I (R0 and R1 20–30 bits)
1001	Two groups of L1OCd and L1OCp were generated in parallel (R0 0–43, R1 0–31)	
1010	Simultaneous generation of three L2OCp or L3OCd, L3OCp (R0 0–41, R1 0–20)	
1011	E5b-I and Q components (R0 and R1 bits 0–27)	
1100	Simultaneous generation of L6D/E (R0 0–39, R1 0–19)	

4. Experimental Results

4.1. FPGA Verification Results

In this paper, the proposed universal code generator structure is implemented using Verilog HDL and simulated using Vivado 2018.3. To test and validate the experimental results, the spread spectrum code generation for all the code generators is implemented using MATLAB R2021a. The generated codes are then compared with the spread spectrum codes produced by the FPGA, confirming the accuracy of the generated codes. Figure 12 illustrates a partial spread spectrum code waveform obtained from the FPGA, where each rising clock edge corresponds to the output of one bit of the spread spectrum code.

4.2. Resource Consumption

In the experiment, synthesis was performed using Xilinx Vivado 2018.3, where the FPGA chip was Zynq-xc7z100ffg900-2. A comparison was conducted between the previous code generator and the code generator proposed in this paper, yielding the following results:

Compared with the previous code generator, the code generator proposed in this paper features a more sophisticated hardware structure. It supports various GNSS signal spread-spectrum codes, including IRNSS IZ4, QZSS L6, and others, based on an LFSR structure. Additionally, it has the capability to generate multiple sets of spread-spectrum codes in parallel, such as simultaneous generation of L5 signal data and pilot channel spread-spectrum codes, or the simultaneous generation of five groups of L1C/A signal spread-spectrum codes and other configurable modes. Table 4 provides a comparison between the number of supported spread-spectrum codes by existing universal code generators and the proposed universal code generator in this paper. The results demonstrate

a significant improvement in the compatibility of the proposed scheme. Figure 13 displays the comprehensive resource utilization of FPGA implementation.

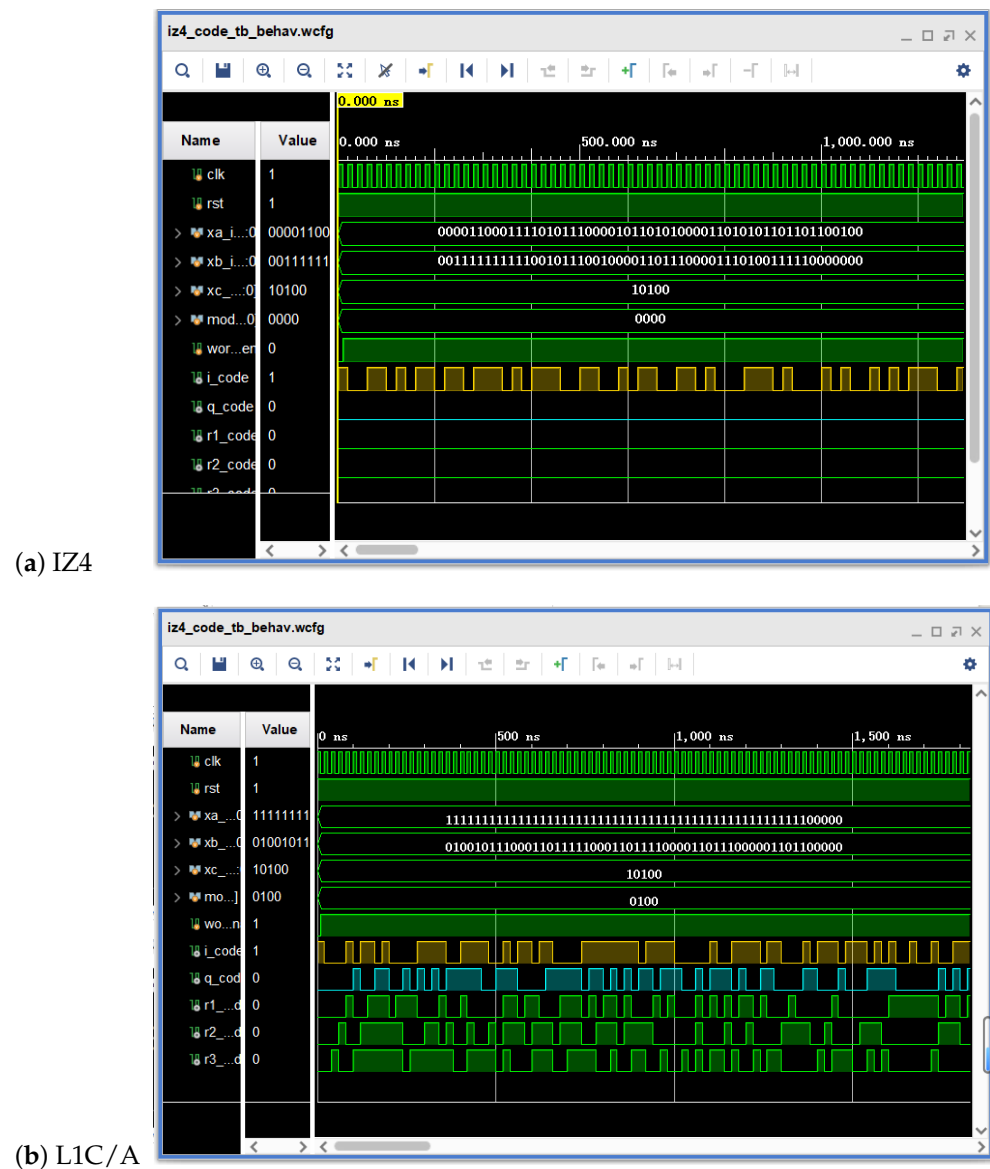


Figure 12. FPGA Simulation Results Based on LFSR.

Utilization	Post-Synthesis			Post-Implementation
				Graph Table
Resource	Estimation	Available	Utilization %	
LUT	385	277400	0.14	
FF	129	554800	0.02	
IO	127	362	35.08	
BUFG	1	32	3.13	

Figure 13. FPGA Comprehensive Resource Results.

Table 4. Comparison of the Number of Signals Supported by This Paper and Previous Code Generators.

Supported Systems		BDS	GPS	GLONASS	Galileo	IRNSS	QZSS	SBAS	Summary
Supported Signals	Previous [6]	①–④ ⑥	⑦ ⑨–⑫	⑬–⑱	⑳–㉑	㉓–㉔	㉖–㉗	㉙–㉚	Table 1 shows the markings
	Proposed	①–⑥	⑦ ⑨–⑫	⑬–⑱	⑳–㉑	㉓–㉕	㉖–㉙	㉙–㉚	

5. Conclusions

This paper analyzes the generation structure of GNSS spread spectrum codes and proposes a universal code generator structure that supports the IZ4 ranging code in the L1-SPS signal newly released by IRNSS in India. It enables the generation of all LSFR-based spread spectrum codes listed in Table 4. By segmenting the structure of the IZ4 code generator, reconstructing the tap selection, and comparing it with the previous universal spread spectrum code generator, this paper improves the utilization and compatibility of the IZ4 code generator. Due to the additional support for IZ4 signals, this paper proposes a more complex universal spread spectrum code generator compared to the previous version. However, it maintains compatibility with various GNSS signal spread spectrum code generation techniques. The proposed code generator can simultaneously generate multiple sets of GNSS signal spread spectrum codes, such as L5 signal data and pilot channel spread spectrum codes, or five groups of L1C/A signal spread spectrum codes, four groups of B3I signal spread spectrum codes, and other mode options. This significantly enhances hardware utilization efficiency.

Author Contributions: Methodology, X.B.; Software, T.L.; Validation, T.L.; Formal analysis, T.L.; Investigation, T.L.; Data curation, T.L.; Writing—original draft, T.L.; Writing—review & editing, X.B.; Visualization, T.L.; Supervision, X.B.; Project administration, X.B., W.J., J.W., B.C., L.C. and K.L.; Funding acquisition, X.B., W.J., J.W., B.C., L.C. and K.L. All authors have read and agreed to the published version of the manuscript.

Funding: This paper is supported by the National Key R&D Program of China (2022YFB4300500) National Natural Science Foundation of China (62027809, T2222015, U2268206), the Talent Fund of Beijing Jiaotong University (No. 2022XKRC003).

Data Availability Statement: The data and the code will be available upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bartunkova, I.; Eissfeller, B. Broadband multi-frequency GNSS signal simulation with GPU. In Proceedings of the 2016 IEEE/ION Position, Location and Navigation Symposium (PLANS), Savannah, GA, USA, 11–14 April 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 477–490.
2. Mikhaylova, O.K.; Korogodin, I.V.; Lipa, I.V. Universal ranging code generator of GLONASS and GPS open navigation signals. In Proceedings of the 2020 International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE), Moscow, Russia, 12–14 March 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–5.
3. Fortin, M.-A.; Landry, R. Implementation Strategies for a Universal Acquisition and Tracking Channel Applied to Real GNSS Signals. *Sensors* **2016**, *16*, 624. [CrossRef] [PubMed]
4. Yang, Y.; Ba, X.; Chen, J. A Novel VLSI Architecture for Multi-Constellation and Multi-Frequency GNSS Acquisition Engine. *IEEE Access* **2019**, *7*, 655–665. [CrossRef]
5. Kim, M.; Park, J.; Jo, G.; Yoo, H. Area-Efficient Universal Code Generator for Multi-GNSS Receivers. *Electronics* **2021**, *10*, 2485. [CrossRef]
6. ISRO. “Satellite Navigation Services.” ISRO—Indian Space Research Organisation. Available online: www.isro.gov.in/ISRO_EN/SatelliteNavigationServices.html (accessed on 23 October 2023).
7. Jiang, Y.; Li, J. Design of BDS/GPS C/A Code Generator. In Proceedings of the 2021 4th International Conference on Electron Device and Mechanical Engineering (ICEDME), Guangzhou, China, 13–19 March 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 5–9.
8. Chityala, H.K.; Nayana, D.K.; Dharmappa, D. PRN sequence design based on Concatenation of Weil Sequences (CWS). In Proceedings of the 2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN), Vellore, India, 5–6 May 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–4.

9. Park, J.; Jo, G.; Kim, J.; Yoo, H. Area-Efficient On-the-Fly Code Generator for BDS B1C Receivers. In Proceedings of the 2021 International Conference on Electronics, Information, and Communication (ICEIC), Jeju, Republic of Korea, 31 January–3 February 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–4.
10. Park, J.; Kim, M.; Jo, G.; Yoo, H. Area-Efficient Universal Code Generator for GPS L1C and BDS B1C Signals. *Electronics* **2021**, *10*, 2737. [CrossRef]
11. China Satellite Navigation Office. BeiDou Navigation Satellite System Signal in Space Interface Control Document Open Service Signal. Available online: http://www.beidou.gov.cn/rdzt/bdwx/ICD_OpenServiceSignal_BDS-ICD-OS-SIS-V1.0.pdf (accessed on 23 October 2023).
12. Global Navigations Satellite System GLONASS. Navigational Radio Signal in Bands L1, L2. Available online: <http://docs.cntd.ru/document/465317671> (accessed on 23 October 2023).
13. U.S. Air Force. Interface Specification IS-GPS-800H. NAVSTAR GPS Space Segment/User Segment L1C Interfaces. Available online: <https://www.gps.gov/technical/icwg/IS-GPS-800L.pdf> (accessed on 23 October 2023).
14. European Space Agency. Galileo Open Service Signal-In-Space Interface Control Document. Available online: <https://www.gsc-europa.eu/sites/default/files/2021-05/GalileoOS-SIS-ICD-V1.3.pdf> (accessed on 23 October 2023).
15. He, X.; Ba, X.; Liu, X. Analysis of QZSS System L6 Signal. In Proceedings of the 11th China Satellite Navigation Conference—S09 User Terminal Technology, Chengdu, China, 22–25 November 2020; Academic Exchange Center of China Satellite Navigation System Management Office: Shanghai, China, 2020; pp. 36–40.
16. Kumar, P.V.; Dharmappa, D.; Mishra, S. Interleaved Z_4 -Linear Sequences with Improved Correlation for Satellite Navigation. In Proceedings of the 2021 IEEE International Symposium on Information Theory (ISIT), Virtual Event, 12–20 July 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1665–1670.
17. Puga, G.L. Linear-feedback shift register seed determination for memory-constrained embedded systems. In Proceedings of the 2017 IEEE URUCON, Montevideo, Uruguay, 23–25 October 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–4.
18. Klein, A. *Stream Ciphers*; Springer: London, UK, 2013.
19. Kasami, T. *Weight Distribution Formula for Some Class of Cyclic Codes*; University of Illinois: Champaign, IL, USA, 1966.
20. Sarwate, D.V.; Pursley, M.B. Crosscorrelation properties of pseudorandom and related sequences. *Proc. IEEE* **1980**, *68*, 593–619. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.