# scientific reports

OPEN

# An effective lossless compression method for attitude data with implementation on FPGA

Fangxing Lyu[1,2,3], Zekang Xiong[1,2,3], Fei Li[1,2,3 ✉], Ying Yue[1,2,3] & Nan Zhang[1,2,3]

The attitude angles of the drilling tool serve as crucial information for transmitting Measurement While Drilling (MWD) data, enabling the optimization of drilling performance and ensuring tool safety. However, the real-time transmission and processing of attitude data pose a significant challenge, especially with the increasing prevalence of horizontal and directional drilling. To accurately and promptly obtain the attitude data, this paper proposes a lossless compression method based on Huffman coding, called Adaptive Frame Prediction Huffman Coding (AFPHC). This approach leverages the slowly varying characteristics of MWD tool attitude data, employing frame residual prediction to reduce data volume and selecting optimal bit widths for encoding transmission data. By using real-world drilling data, the proposed method is implemented on a Verilog HDL on a Xilinx field-programmable gate array (FPGA) circuit. Simulation and experiment results show that compression ratios provided by the proposed method for the inclination, azimuth, and toolface angles reach up to 4.02 times, 3.98 times, and 1.48 times, respectively, outperforming several existing methods.

The Rotary Steerable System (RSS), the most effective drilling technology for unconventional oil, gas, and new energy development to date, offers advantages such as high drilling speed, low cost, and smooth wellbore trajectories[1,2]. As a component of RSS, attitude measurement is a crucial prerequisite in enabling directional control[3]. Mud Pulse Telemetry (MPT) is the most widely used data transmission technology for MWD, but its transmission rate is only 2-5 bits/s[4–6]. With the continuous advancement of MWD technology, there is an increasing demand for real-time transmission and higher accuracy of drill tool attitude data. This has led to a significant increase in the amount of attitude data, which poses a great challenge for transmitting and processing attitude data[7]. Data compression technology solves this issue and realises timely transmission of while-drilling measurement data[8]. Two primary techniques, lossy and lossless, have been developed in the data compression technology to MWD engineering[9,10].

## Lossy data compression techniques

Lossy data compression techniques achieve higher compression ratios by discarding some data, causing data accuracy or quality to reduce[11]. In MWD data transmission, differential pulse code modulation (DPCM) and wavelet transform are the main techniques for lossy data compression[12,13]. For the former, an improvement to DPCM has been proposed by combining non-uniform quantization with root mean square (RMS) propagation gradient descent, achieving compression ratios of 2.73 and 3.06 times for resistivity and gamma data, respectively[14]. A compression algorithm for underground remote acoustic logging data was designed based on adaptive differential pulse modulation, achieving a 50% data compression ratio while maintaining a distortion of only 3%[15]. For the latter, a downhole video image compression system based on wavelet transform is proposed. By changing the compression factor, some image information is lost, resulting in a data compression ratio greater than 4 times[16]. Photographic images achieved a compression ratio of 1.01 times, while non-photographic images achieved a compression ratio of 1.31 times, both achieved using discrete wavelet transform and prediction methods, respectively[17]. Ling Kaixuan et al. proposed a data rearrangement method to enhance the compression efficiency of a hybrid coding approach based on DPCM and Discrete Cosine Transform (DCT). Under the premise of keeping the reconstruction error within 5%, they achieved a 60% improvement in the compression ratio of imaging-while-drilling data[18]. Zhang et al. proposed a quantized Compressed Sensing (CS) method

[1]Xi'an Key Laboratory of Intelligent Equipment Development for Oil, Gas and Renewable Energy, Xi'an Shiyou University, Xi'an 710065, Shaanxi, China. [2]Directional Drilling Laboratory of CNOOC Key Laboratory of Well Logging and Directional Drilling, Xi'an Shiyou University, Xi'an 710065, Shaanxi, China. [3]School of Electronic Engineering, Xi'an Shiyou University, Xi'an 710065, Shaanxi, China. ✉email: lif@xsyu.edu.cn

that achieves a peak signal-to-noise ratio of approximately 20 dB at an extremely low sampling rate of 1% on real-world datasets[19]. Wolfgang Weinzierl et al. developed a PCA-based method combined with a convolutional autoencoder to compress real-time NMR downhole data of three echo trains to fewer than 100 bits, addressing telemetry constraints in drilling operations[20]. However, lossy compression results in some distortion, preventing the compressed data from being fully restored to its original form[21]. For real-time attitude data, this can lead to inaccuracies in data calculations, reducing the reliability of the data.

## Lossless data compression techniques

Lossless data compression techniques reduce the data volume by eliminating redundancy and repetitive information, which benefits the integrity and accuracy of the data[22]. The classic compression methods include run-length coding, arithmetic coding, dictionary-based coding, Lempel-Ziv-Welch (LZW) coding, Huffman coding, etc.[23]. For MWD data, the correlation between different physical quantities is weak, and there is a lack of a large number of repeated characters, causing run-length coding to be ineffective[24]. Despite its high compression ratio, arithmetic coding is unsuitable for compressing MWD data due to the difficulties in its hardware implementation[25]. Dictionary-based coding can improve the compression ratio, but it introduces system latency, which impacts real-time performance[26]. Therefore, the main methodologies applied in the lossless compression of MWD data are the LZW and the Huffman coding algorithms. For the former, Su et al. proposed an LZW data compression system based on Minimum Edit Distance prediction, achieving a compression ratio of 1.42 for sonic logging data[27]. After that, a joint algorithm of LZW lossless compression and Reed-Solomon coding was presented to encode and compress mud channel data, achieving a compression ratio of 2.75 in experiments[28]. For the latter, Shan Song et al. proposed a grouped frame prediction Huffman coding method and compared it with LZW, Huffman, and adaptive Huffman methods, among other lossless compression algorithms. They achieved a compression ratio of 1.41 for the compression of logging-while-drilling data[29]. Thereafter, a mixed coding algorithm based on adaptive Huffman and Golomb-Rice was proposed, achieving a compression ratio of 4.11 for wireless sensor network data[30]. However, this kind of coding table based on real-time statistical features makes it difficult to recover the original data in case of transmission errors, which makes it unsuitable for MPT[31]. A combined coding method using DCT, run-length coding, and Huffman coding was proposed for MWD nuclear magnetic resonance echo data, achieving a compression ratio of 15 with a relative error of less than 5%[32]. Although this method achieves a relatively large compression ratio, it requires the target data to have obvious continuous repetitiveness and multi-exponential variation features, making it unsuitable for MWD attitude data[33]. Chen Jianhua et al. proposed a deep learning-based lossless data compression method for well-logging data, achieving a 23% improvement in compression ratio for one-dimensional well-logging data and a 21% improvement for two-dimensional well-logging data[34]. Shan Song et al. employed a compression method utilizing deep autoencoders, which reduces errors by compressing and transmitting residual data from the feature extraction process through quantized encoding and Huffman coding, achieving a data compression ratio of 4.05 for inclination and azimuth angles[35]. Despite achieving high compression ratios through advanced artificial intelligence algorithms, the practical implementation of these techniques in downhole drilling tools remains challenging.

The literature review of the aforementioned compression methods for logging-while-drilling data is summarized in Table 1. The combination of various lossless compression algorithms requires high processing time and computing resources. In terms of algorithms, compared with joint algorithms, the Huffman coding algorithm is simple to operate and highly feasible, making it more suitable for MWD data processing[36]. The algorithm reduces the encoded data size by constructing an optimal prefix code based on character frequency. It dynamically adjusts the code length according to the frequency of characters, achieving effective data compression. Although Huffman coding has been applied to compress MWD data, it can only achieve a compression ratio of 1.2 to 1.8[37]. In the current field of lossless compression for MWD data, Huffman coding has yet to be effectively improved or optimized. In terms of hardware implementation, due to the limited space of downhole drilling tools and the complex geological environment, the high integration and low power consumption characteristics of the hardware are put forward strict requirements. FPGAs offers high integration, low power consumption, flexibility, and parallel computation capabilities with simpler algorithm implementation[38–40]. Reconfigurable programming also facilitates updating the Huffman coding table to adapt to new geological environments. Therefore, FPGA circuits serve as an ideal embedded hardware development platform for attitude data compression[41–43].

A high compression ratio often means sacrificing data accuracy while-drilling in engineering applications[44,45]. However, the accuracy requirements for MWD attitude data are particularly strict. To address this challenge, improving the data compression ratio while maintaining the required accuracy of MWD attitude data to reduce data transmission volume has become an urgent issue in engineering applications.

## Contributions

To address the problem of timely and accurate transmission of attitude data while-drilling, this paper develops an AFPHC algorithm to realize lossless compression of attitude data. The method relies on the slowly varying data characteristics of the attitude data, performing frame residual prediction to reduce data transmission volume. Based on the predicted residual features, we propose an improved adaptive-range Huffman coding method to select the appropriate bit width for encoding the transmission data, thereby enhancing the efficiency of attitude data coding and bandwidth utilization. By building an attitude data compression system while-drilling based on FPGA, it also leverages the advantages of parallel computation, easy embedding, and low power consumption of FPGA to improve the real-time performance of attitude data transmission.

| Ref@year | Main methods | data object | Compression ratio | Method type |
|---|---|---|---|---|
| [14]@2023 | DPCM | Resistivity data | 2.73 | Lossy |
| | | Gamma data | 3.06 | |
| [15]@2023 | Adaptive DPCM | Acoustic logging data | 2.00 | |
| [16]@2007 | Wavelet transform | Logging image | 4.00 | |
| [17]@2020 | Wavelet transform & prediction | Photographic image | 1.01 | |
| | | Non-photographic image | 1.31 | |
| [18]@2018 | DPCM & DCT | Imaging while drilling data | 2.50 | |
| [32]@2018 | DCT & RLE & Huffman | Nuclear magnetic resonance data | 15.00 | |
| [19]@2024 | CS | Ultrasonic image | Effectual | |
| [20]@2024 | PCA | Nuclear magnetic resonance data | Effectual | |
| [27]@2015 | MED & LZW | Acoustic logging data | 1.42 | Lossless |
| [28]@2016 | LZW & RS | Mud channel data | 2.75 | |
| [29]@2021 | FPHC | MWD logging data | 1.41 | |
| | Huffman | | 1.11 | |
| | Adaptive Huffman | | 1.08 | |
| | LZW | | 0.83 | |
| [30]@2016 | Huffman & Golomb-Rice | Wireless sensor network data | 4.11 | |
| [34]@2021 | Deep learning | One-dimensional logging data | 1.29 | |
| | | Two-dimensional logging data | 1.26 | |
| [35]@2024 | Depth automatic coding | Inclination and Azimuth data | 4.05 | |

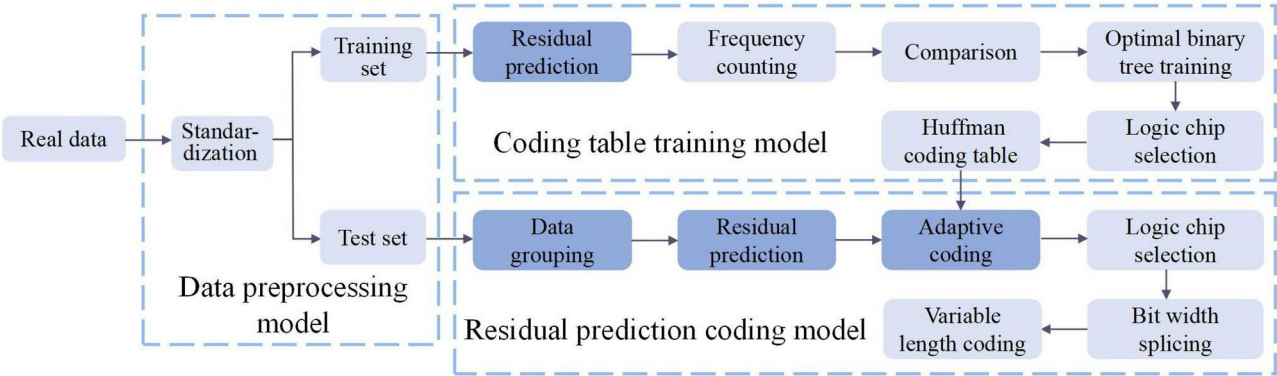**Table 1**. Literature Review of Downhole Logging Data Compression Methods.



**Fig. 1**. Framework of drill tool attitude data compression system.



**Fig. 2**. Data frame format.

## Method and implementation
### System framework and key technologies

The FPGA-based attitude data compression system is depicted in Fig. 1. It primarily comprises the following models: data preprocessing model, coding table training model, and residual prediction coding model. The data preprocessing model standardizes the logging file data, converting the attitude angle data into frame-format data. The data frame format of one frame of logging data is shown in the following Fig. 2. The attitude angle data in the logging file is in decimal form $M$, which needs to be converted into binary data $C$ according to the protocol of uploading coded data in the rotary steering system $C = \frac{M}{P}$, where $P$ is the physical quantity versus accuracy. For example, the azimuth is $188.8°$, the corresponding precision $P$ is $0.17578125$, and the coded value is 11-bit binary coded value and 1-bit even parity bit, that is, 100001100100.

The coding table training model applies Huffman algorithm coding training to the frame-format data of the training set, generating a fixed probability coding table. The main steps involve: counting the character frequency, establishing an optimal binary tree based on the character frequency, and sequentially outputting the codes corresponding to the characters by logic chip selection.

The residual prediction coding model performs residual prediction between frames on the frame-format data of the test set. It generates the adaptive range Huffman coding output for the predicted residuals based on the fixed probability coding table. This section will mainly introduce the improved and innovative algorithms in this these models. The main steps include: grouping data frames, carrying out residual prediction on the data in the frame group, applying Huffman coding to the prediction residual with self-adaptive bit width, selecting and outputting coding on the logic chip of the prediction residual character, and carrying out bit splicing to realize the coding output of each frame one by one.

In Fig. 1, data grouping, residual prediction and adaptive coding are the core modules and technical difficulties, which will be introduced in the following sections.

*Frame grouping and packing method*
Since the output of Huffman coding method is bitstream data, it has relatively poor error correction ability and is susceptible to channel errors caused by noise interference. Because the flexible grouping method can prevent error spreading and improve its error resilience without affecting the compression ratio of Huffman coding, the robustness of the system can be improved by using grouping and packaging. The main principle is as follows:

(1)  Independent grouping: Group all frames in the logging file. Each frame group is independent so as to prevent error propagation.
(2)  Synchronized processing: The number of frames in each frame group is the same and fixed, ensuring synchronization between the encoder and decoder.

This paper uses grouping parameters $K = 10$ as an example for data packing without loss of generality. However, the grouping parameters $K$ can be optimized in drilling operations based on the actual geological environment.

The packet packing state machine realized by FPGA is shown in Fig. 3, which mainly includes five states: initial state *Idle* to realize global reset, frame header search state *Header_search*, frame data acquisition state *Frame_receive*, frame tail check state *Tail_check* to realize data reception integrity judgment, and frame group packing state *Packet_form*.

*Residual prediction method*
There are a total of $Z$ data frames in the logging data file. When processing the current $i - th$ data frame, the $i - th$ frame is defined as the current frame, and the $(i - 1) - th$ frame is defined as the predicted reference frame.

In the MWD data file, the frames are grouped into sets of $K$ frames, referred to as frame groups. In each frame group, the data frames are labeled sequentially according to the acquisition time, denoted as $F_0, F_1, F_2, ..., F_{K-1}$, i.e., Frame 0, Frame 1, Frame 2, ..., Frame $K - 1$. A frame group can be represented as:

$$F_i = \left\{ \begin{array}{l} \{B_{0,I}, \ldots, B_{0,A}, \ldots, B_{0,T}\} \\ \{B_{i,I}, \ldots, B_{i,A}, \ldots, B_{i,T}\} \\ \{B_{K-1,I}, \ldots, B_{K-1,A}, \ldots, B_{K-1,T}\} \end{array} \right. , \tag{1}$$

where $I$ is the inclination angle, $A$ is the azimuth angle, and $T$ is the toolface angle. In order to facilitate calculation, all characters in the reference frame of frame 0 are initialized to 0. Treat all $B_{i,j}(j = I, A, T)$ as unsigned integers, then compute the prediction residuals between the corresponding physical quantities of the current frame and its reference frame.

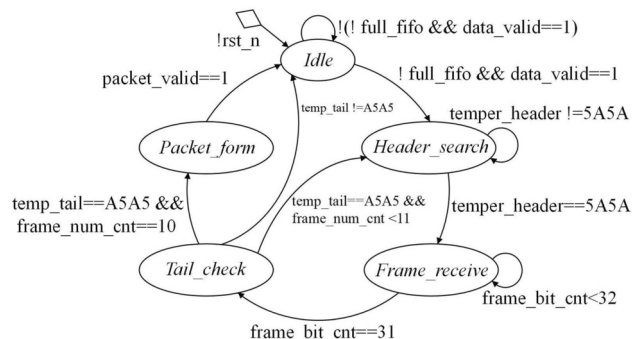The prediction residual can be expressed as:



**Fig. 3**.  Packet packing state machine.

$$R_{i,j} = \begin{cases} e_{i,I} = B_{i,I} - B_{i-1,I} \\ e_{i,A} = B_{i,A} - B_{i-1,A} \\ e_{i,T} = B_{i,T} - B_{i-1,T} \end{cases}, \qquad (2)$$

where $e_{i,I}, e_{i,A}, e_{i,T}$ are signed integers with value ranges of $-2047 \le e_{i,I}, e_{i,A} \le 2047$ and $-63 \le e_{i,T} \le 63$, and the value range of $i$ is $0 \le i \le K-1$.

The principle is shown in Fig. 4. The FPGA timing circuit design uses chip select signal of $CS$ control to implement three-state gating, ensuring that only one byte of data from the same frame is received on the same clock-rising edge. The enable signals of $en\_W_1$, $en\_W_2$, and $en\_W_3$ inputs control the data writing. Before the current frame register holds the next frame data, the data of the current frame is transferred to the predictive frame register, iteratively, completing the residual prediction between frames for the frame group.

*Adaptive-range Huffman coding method*

Adaptive prediction residual range [-9, 9] encoding better aligns with attitude data characteristics, while the encoding scheme maintains low FPGA logic resource consumption for practical implementation. The distribution of a small number of residual prediction characters is scattered, and the high-order bits of the corresponding encoded data are filled with many zeros, occupying bit-width resources. Based on this feature, this paper proposes an adaptive-range Huffman coding method for predicting residuals between attitude data frames. The procedure is as follows: (1) Residual values of the drilling attitude angle prediction within the range of $[-9, 9]$ are encoded and transmitted using Huffman coding. (2) Residual values of the drilling attitude angle prediction within the ranges of $[-63, -10]$ and $[10, 63]$ are encoded using a fixed bit-width of 7 bits (1 bit for the sign and 6 bits for the data). (3) The predicted residual values of the inclination and azimuth are transmitted in a fixed coded bit width of 12 bits (1 bit for the sign and 11 bits for the data) in other ranges.

$$e_{i,I} = e_{i,A} = \begin{cases} \text{Indefinite length} & \text{for } R_{i,j} \in [-9, 9] \\ 7bit & \text{for } R_{i,j} \in [-10, -63] \cup [10, 63] \\ 12bit & \text{for } R_{i,j} \in [-64, -2047] \cup [64, 2047] \end{cases}. \qquad (3)$$

$$e_{i,T} = \begin{cases} \text{Indefinite length} & \text{for } R_{i,T} \in [-9, 9] \\ 7 \text{ bit} & \text{for } R_{i,T} \in [-10, -63] \cup [10, 63] \end{cases}. \qquad (4)$$

## FPGA implementation

Digital RTL synthesis and information theory achieve synergistic optimization through functional complementarity, with the FPGA architecture providing a unified hardware implementation platform. As the algorithm requires specific logic resources and IP cores for FPGA, this paper selects the Artix-7 series FPGA chip XC7A100T-2FGG484I from Xilinx, USA, for the system-level circuit design. The FPGA logic design, project configuration, environmental compilation, and timing simulation are carried out on the Vivado 2023.2 software[46]. During the experimental verification process, the proposed algorithm was embedded into the FPGA for functional validation via software simulation. The realization of the program includes the following steps: The host computer uses Qt Creator 12.0.2 software[47] to read the attitude data and convert it into frame-format files. The frame-format data is then sent to the FPGA circuit via USB-to-serial communication. The input data is processed in parallel to implement the algorithm's logic functions inside the FPGA. Finally, the encoded bitstream data is output. The encoded data is transmitted to the agent device according to the UART communication protocol, where the slave computer's Python 3.11.0 software[48] decodes the data and displays the attitude angles. The algorithm is implemented on FPGA as shown in Fig. 5.

Finally, the attitude angle data displayed by the slave computer is shown in Fig. 6. It mainly includes: the serial port settings module, data transmission module, data reception module, and attitude angle display disk module.
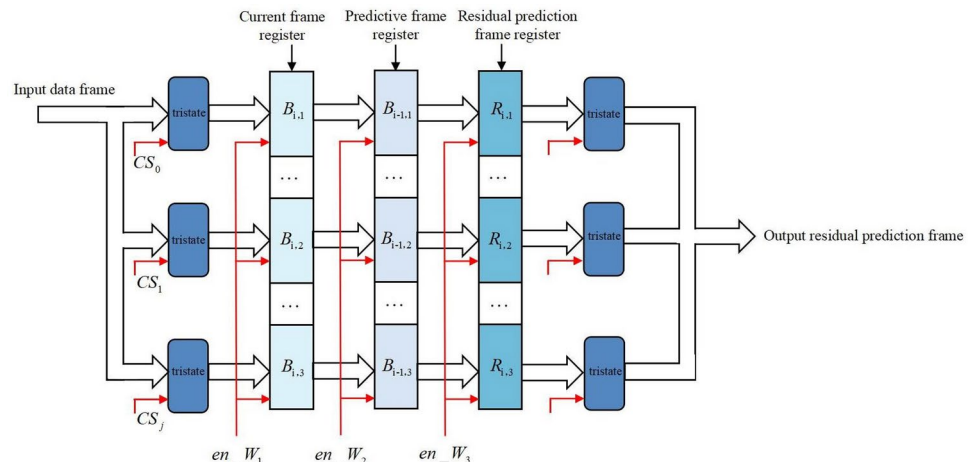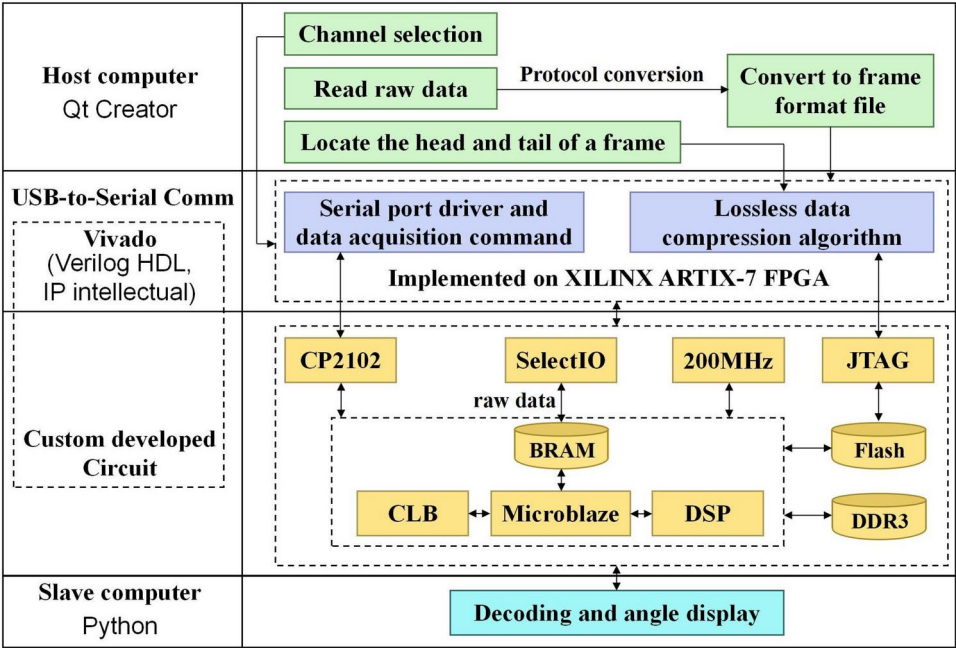


**Fig. 4.** Bock diagram of residual prediction frames.

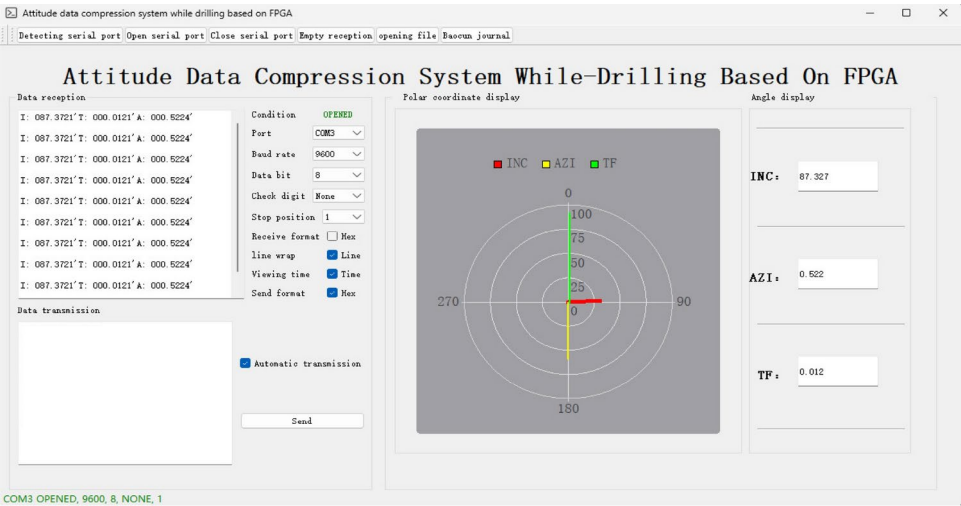**Fig. 5.** Block diagram of the proposed algorithm implemented on FPGA.



**Fig. 6.** Receiving data display interface of slave computer.

The serial port settings module is used to configure the baud rate and parity settings. The data transmission module is responsible for sending the attitude data. The data reception module receives the encoded data sent by the FPGA, performs decoding and decompression, and calculates the attitude angle frame format. The disk interface is primarily used to display the drill tool's attitude angles, enabling the visualization of the drill tool's attitude.

The lossless compression method of attitude data while-drilling based on residual feature prediction coding runs on an FPGA circuit. Its thermal margin is 58.5 °C, and the total power consumption is 0.87 W, which meets the low-power consumption requirement of MWD drilling tools. The compression algorithm circuit described in this paper utilizes only a limited amount of hardware resources within the FPGA chip as Table 2 shows. By efficiently mapping the operation sequence of the FPGA parallel architecture, it avoids excessive control logic or memory dependencies.

## Method validity analysis

To effectively compress MWD attitude data, this study collected eight logging data files obtained from the actual drilling process in Luntai Oilfield, Xinjiang, China. The logging files have been divided into two

| Performance index | Resource usage | Resource utilization |
|---|---|---|
| Look-Up Tables | 29788 | 35.80% |
| Flip-Flops | 32182 | 21.72% |
| Digital Signal Processors | 30 | 11.36% |

**Table 2**. Resource of algorithm on FPGA circuit.

groups: $Group_1$ and $Group_2$. $Group_1$, as the training set, contains four files: 1st, 2nd, 3rd, and 4th, with each containing more than 50,000 frames of measurement data, used for preliminary feature analysis and training the Huffman probability table. $Group_2$, as the test set, includes four files: 5th, 6th, 7th, and 8th, used for algorithm simulation and testing experiments. This section establishes the original character probability model and the predicted residual probability model. Normal distribution curve fitting analysis is performed to demonstrate the effectiveness of the residual prediction method.

Each data frame in the logging files is treated as three independent decimal characters, $a_i$, representing the physical quantities of inclination angle, azimuth angle, and toolface angle. According to Shannon's information theory[49], the entropy of the random variable $H(A)$ is given by:

$$H(A) = -\sum_{i=1}^{n} p(a_i) \cdot \log p(a_i), \tag{5}$$

where $p(a_i)$ is the probability of the $i-th$ character in the logging data file. According to Shannon's first theorem[50], the entropy $H(A)$ of each character in the logging file represents the theoretical limit of the binary code length for lossless compression, i.e.,

$$H(A) = -\sum_{i=1}^{n} p_i \cdot \log p_i \leq -\sum_{i=1}^{n} p_i \cdot \log q_i. \tag{6}$$

The equality holds only when $\{q_i\} = \{p_i\}$, with $0 \leq q_i \leq 1 (j = 1, 2, ..., n)$ and $\sum_{i=1}^{n} q_i = 1$. The average code length $L$ of all character $a_i$ codes in the logging file is:

$$L = \sum_{i=1}^{n} p_i \cdot l_i = -\sum_{i=1}^{n} p_i \cdot \log q_i, \tag{7}$$

where $l_i$ is the coding length of the character $a_i$, which is a non-negative random variable with $l_i = -\log q_i$. According to the maximum discrete entropy theorem[51], the entropy of all probability distributions $p_i$ is the maximum when the probability is equal, that is:

$$H(\mathrm{p}) = H_n(p_1, p_2, \cdots, p_n) \leq \log n, \tag{8}$$

where $\mathrm{P} = (p_1, p_2, \cdots, p_n)$ is the n-dimensional probability vector. $n = 2^R$, in which $R$ is the binary coded bit width. The average information redundancy contained in all characters in the logging file is then given by:

$$r = H_{\max}(A) - H(A) = \log n - H(A). \tag{9}$$

The key to achieving lossless compression of drill tool attitude data lies in reducing the information redundancy of the corresponding physical quantities in the drilling measurement data. As long as the probability distribution of all characters in the logging file is not uniform, there is potential for data compression.

### Independent character probability model

Calculate the relationship between characters and probability distributions in each logging file of $Group_1$, and establish character probability models for physical quantities such as inclination, azimuth, and toolface angles. Fit each original probability model to a Gaussian distribution, characterized by the mean $\mu_0$ and the standard deviation $\sigma_0$, which quantifies the data dispersion.

In Fig. 7a–c represent the probability distribution models of the original data characters for the inclination angle, azimuth angle, and toolface angle, respectively. According to Eqs. (7) and (8), entropy has extremality. In the graph, each dataset's $L$ value takes the minimum value, which is the entropy value, calculated using Eq. (6). The probability distribution range of the original characters varies across different logging files, resulting in different $L$ values. (1) The average code length $L$ of the inclination angle and azimuth angle in the 1st and 2nd well logging files is 6.66, calculated from Eq. (7), which is unsuitable for compression. In the 3rd and 4th well logging files, the character probability distribution is more concentrated, resulting in a smaller average code length. (2) The fixed code length $L$ for the toolface angle transmitted in the data frame is 7 bits, with a minimum average code length of 4.978. The character probability distribution is dispersed, making it unsuitable for compression.
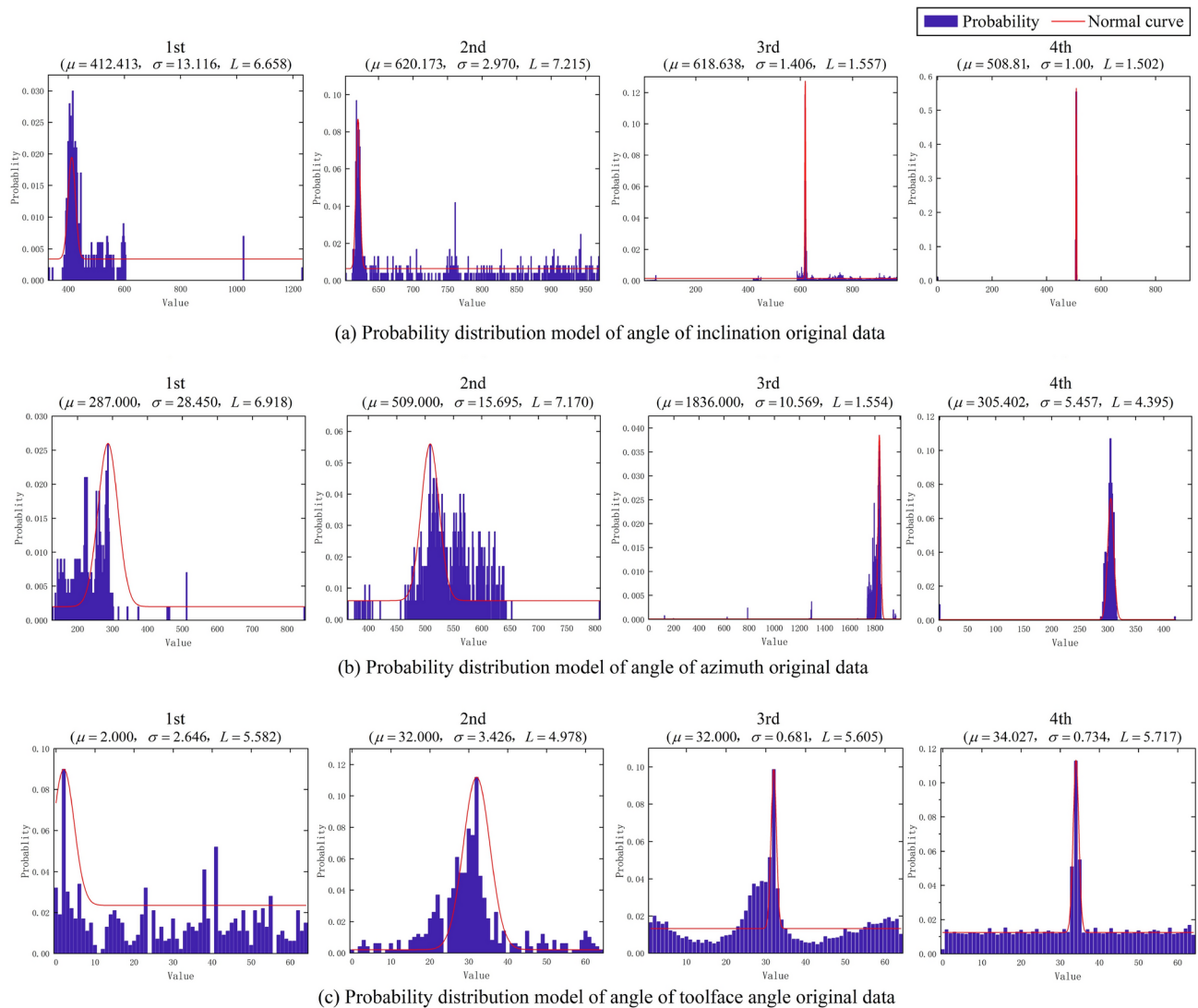
(a) Probability distribution model of angle of inclination original data



(b) Probability distribution model of angle of azimuth original data



(c) Probability distribution model of angle of toolface angle original data

**Fig. 7**. The probability distribution of characters for $Group_1$.

## Predictive residual character probability model

Next, the residuals between frames are predicted for each logging file in $Group_1$. Figure 8 shows the probability model of the predicted residual characters corresponding to the three physical quantities of inclination, azimuth, and toolface angle of the four files. The mean $\mu_p$, standard deviation $\sigma_p$, and average code length $L$ of the Gaussian-distributed random variables for each residual prediction model.

From Fig. 8, it can be revealed that most of the prediction residuals are concentrated in the range of $[-9, 9]$. This suggests that small-range residual character coding is more appropriate for this data. Comparing the variation characteristics of the residual data probability models, the statistical fitting parameters of the normal distribution curve are shown in Table 3.

From Table 3, it can be concluded that:

(1) The mean value $\mu_0$ of the original data probability model ranges from a minimum of 2 to a maximum of 1836, with a relatively discrete distribution. The mean value $\mu_p$ of the residual data probability model is concentrated around 0. As shown in Fig. 8, the probability distribution of the predicted residuals is consistent with the fit of the normal distribution. Therefore, the predicted residual characters are more suitable for data compression than the original ones.

(2) For both inclination angle and toolface measurements, the standard deviation values show a substantial reduction ($\sigma_0 >> \sigma_p$) through the residual prediction process. This reduction demonstrates a clear trend from dispersed to centralized data distribution. The mean value of standard deviation of toolface angle $\sigma_p$ tends to increase compared with $\sigma_0$. The reason is that since the toolface angle fluctuates significantly, the probability distribution of the predicted residuals is more dispersed and uneven.
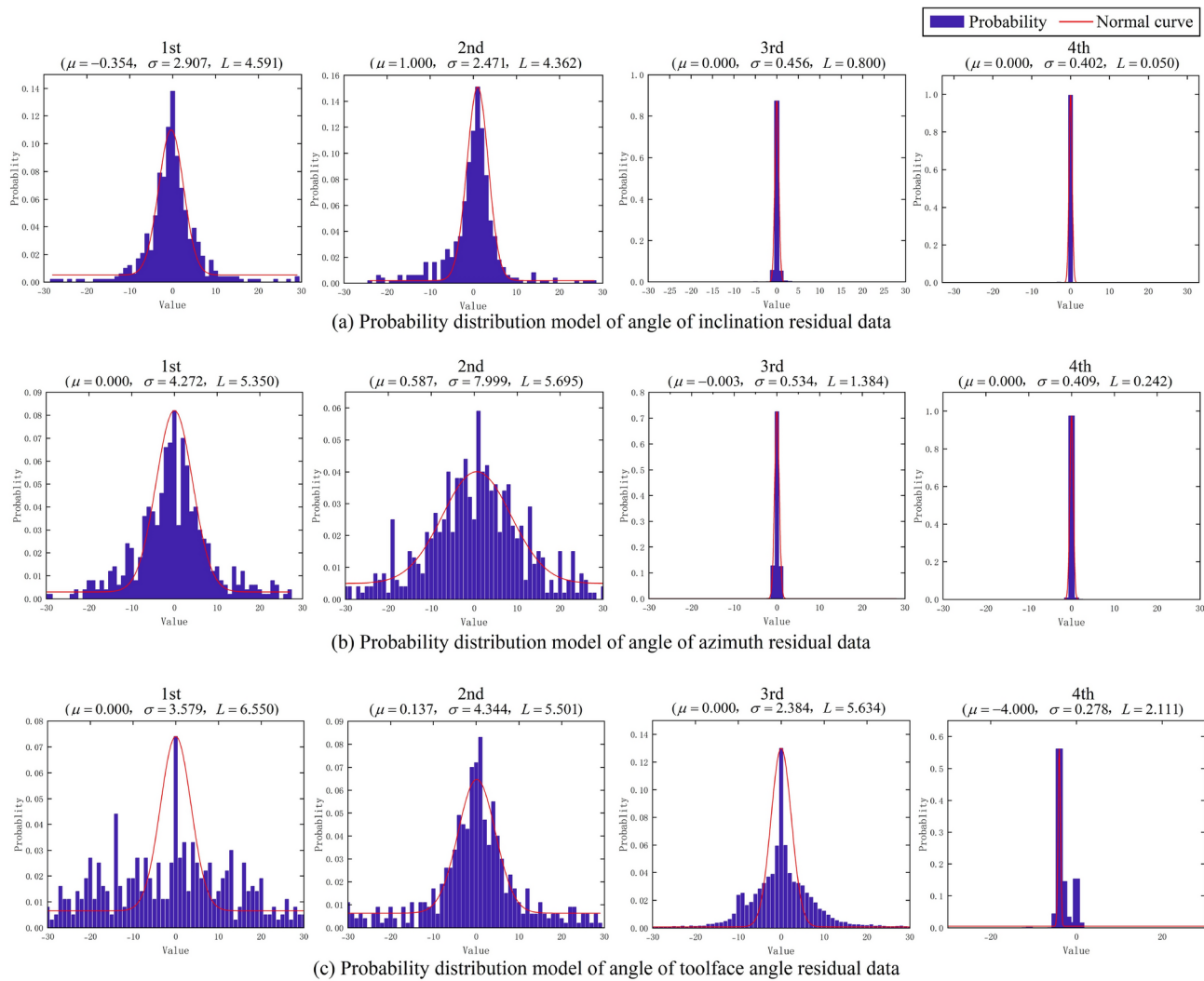
(a) Probability distribution model of angle of inclination residual data

(b) Probability distribution model of angle of azimuth residual data

(c) Probability distribution model of angle of toolface angle residual data

**Fig. 8**. The probability distribution of prediction residual for $Group_1$.

| Parameter | Attitude | 1st | 2nd | 3rd | 4th | $\overline{\mu}$ | $\overline{\sigma}$ |
|---|---|---|---|---|---|---|---|
| $\mu_0$ | INC | 412.413 | 620.173 | 618.638 | 508.811 | 540.009 | 99.771 |
| | AZI | 287.000 | 509.000 | 1836.000 | 305.402 | 734.351 | 741.290 |
| | TF | 2.000 | 32.000 | 32.000 | 34.027 | 25.007 | 15.368 |
| $\sigma_0$ | INC | 13.116 | 2.970 | 1.406 | 0.998 | 4.623 | 5.726 |
| | AZI | 28.450 | 15.695 | 10.569 | 5.457 | 15.043 | 9.867 |
| | TF | 2.646 | 3.426 | 0.681 | 0.734 | 1.872 | 1.382 |
| $\mu_p$ | INC | −0.354 | 1.000 | 0.000 | 0.000 | 0.612 | 0.583 |
| | AZI | 0.000 | 0.587 | −0.003 | 0.000 | 0.146 | 0.294 |
| | TF | 0.000 | 0.137 | 0.000 | −4.000 | −0.966 | 2.024 |
| $\sigma_p$ | INC | 2.907 | 2.471 | 0.456 | 0.402 | 1.559 | 1.317 |
| | AZI | 4.272 | 7.999 | 0.534 | 0.409 | 3.304 | 3.607 |
| | TF | 3.579 | 4.344 | 2.384 | 0.278 | 2.646 | 1.773 |

**Table 3**. Normal Distribution Curve Fitting Parameters.

The average coding length and redundancy of the two models are shown in table 3, where $L_1$, $L_2$, $L_3$ and $L_4$ respectively represent the code lengths of file 1st, 2nd, 3rd and 4th, $\overline{L}$ represents the average code length of files, $\overline{\sigma}$ represents the average standard deviation and $\overline{r}$ represents the average redundancy calculated from Eq. (9).

Table 4 shows that the probability distribution of data is more concentrated and the redundancy $\overline{r}$ of data is increased after residual prediction. The average code length $\overline{L}$ of the characters for the drilling attitude angles decreases, indicating that the residual prediction between frames can effectively improve the data compression performance.

## Results

In 2021, Song et al. used an inter-frame residual prediction method to compress LWD data, achieving an average data compression ratio of 1.41[29]. However, this method has not been implemented on microprocessors such as DSP. Based on the characteristics of attitude data while-drilling, this paper's proposed method improves the adaptability of coded data bit width, so the compression effect is better. And it is deployed on FPGA, which is more in line with the actual engineering needs.

This paper uses the data compression ratio as the indicator of the algorithm's reasonable and effective compression performance. The larger the compression ratio, the greater the improvement in the equivalent transmission rate of the data compression system. The calculation method for the compression ratio in this paper is as follows:

$$C_R = \frac{M}{N}. \tag{10}$$

Where $M$ is the number of bits in the original MWD data file, and $N$ is the number of bits in the MWD data after compression by the FPGA.

### Simulation results and analysis

Following the coding table training method outlined in Fig. 1, the well-logging data files "1st" and "2nd" from set $Group_1$ are read for training to obtain the probability coding table $T_1$ for the residual range $[-9, 9]$; well-logging data files "3rd" and "4th" from set $Group_1$ are used to train and obtain the probability coding table $T_2$ for the residual range $[-9, 9]$.

Figure 9 shows the Huffman coding table $T_1$ output bitstream as an example. It comprises two distinct components: (a) a comprehensive schematic diagram and (b) an enlarged view of the region highlighted by the red circle. The system operates with $clk\_out50$ as the 50 MHz clock signal and $sys\_rst\_n$ as the global reset signal. The dataset, stored in txt format as digital signals, is processed through a sequential readout mechanism. Specifically, the $tx\_data$ signal, with a width of 63 bits and depth of 58,000, systematically retrieves data from the text file and transmits it to the $rx\_data$ signal for subsequent encoding table training. Key signals: $temp\_data$ is fixed-length encoding, and $temp\_len$ is variable-length character encoding size, both used for subsequent logic selection. The $encode\_data$ signal outputs the encoding table in the range of -9 to 9 through a bit-stream format.

Based on this Huffman coding table $T_1$, the attitude data of the 5th well logging file in set $Group_2$ is tested. The bit-stream output of variable-length coding is illustrated in Fig. 10, where $clk$ represents the 50 MHz clock signal and $sys\_rst\_n$ denotes the reset signal. Figure 10a presents the global schematic diagram, showing the processing of the dataset stored in txt format as digital signals through a sequential readout mechanism. Specifically, the $tx\_data$ signal, with a width of 63 bits and a depth of 58,000, systematically retrieves data from the text file and transmits it to the $rd\_data$ signal for subsequent inter-frame residual prediction. Figure 10b provides an enlarged view of the region marked by the red circle in Fig. 10a. The enable signal $en\_flag$ outputs a high-level pulse, indicating the output within a frame of residual prediction signals. The residual prediction frame signal $encode\_data$ is sequentially output through bit-stream encoding. As evident from the figure, the residual prediction frame bit-stream encoding is output in sequence. Each data frame consists of a frame header (5A5A), residual prediction coding, and a frame tail (A5A5).

### Experimental results and analysis

The process of attitude data of 5th logging file from compression to restoration is shown in Fig. 11, which intuitively illustrates the entire sequence of character feature transformations in digital signal processing. Figure 11a shows the real drilling attitude trajectory, from which the attitude angle change information of angle of inclination, azimuth and toolface angle can be obtained. In Fig. 11b, the inclination angle demonstrates an

| Data type | Attitude | $L_1$ | $L_2$ | $L_3$ | $L_4$ | $\overline{L}$ | $\overline{\sigma}$ | $\overline{r}$ |
|---|---|---|---|---|---|---|---|---|
| Fig. 7 | INC | 6.658 | 7.215 | 1.557 | 1.502 | 4.233 | 3.130 | 7.767 |
| | AZI | 6.918 | 7.170 | 1.554 | 4.395 | 5.009 | 2.622 | 6.991 |
| | TF | 5.582 | 4.978 | 5.605 | 5.717 | 5.471 | 0.334 | 1.529 |
| Fig. 8 | INC | 4.591 | 4.362 | 0.800 | 0.050 | 2.451 | 2.361 | 9.549 |
| | AZI | 5.350 | 5.695 | 1.384 | 0.240 | 3.168 | 2.762 | 8.832 |
| | TF | 6.550 | 5.501 | 5.634 | 2.111 | 4.949 | 1.949 | 2.051 |

**Table 4.** Comparison of Parameter Average Code Length.
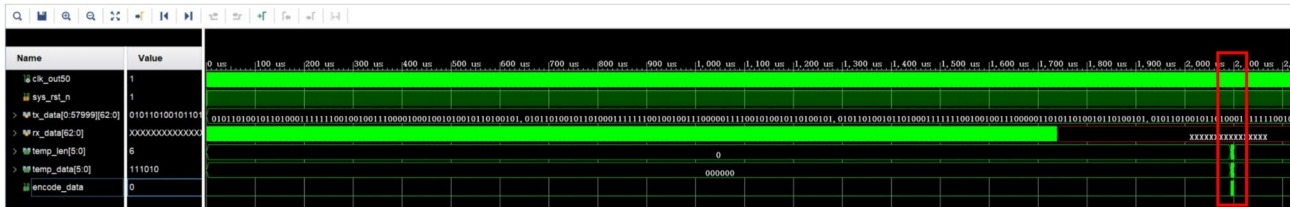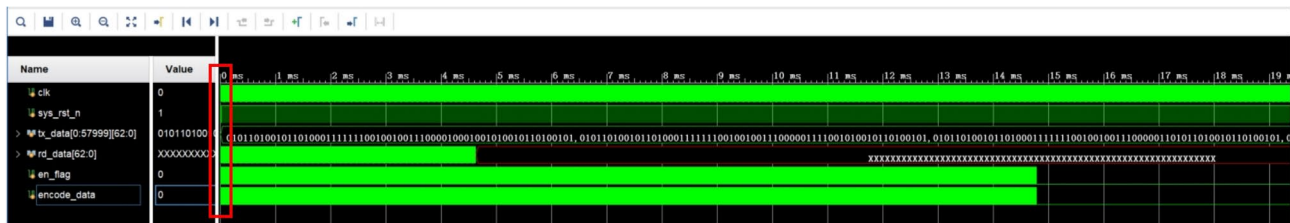
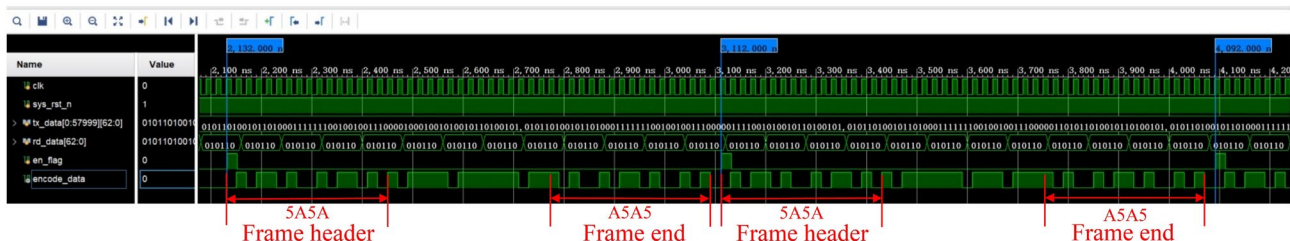(a)  Simulation Global Map
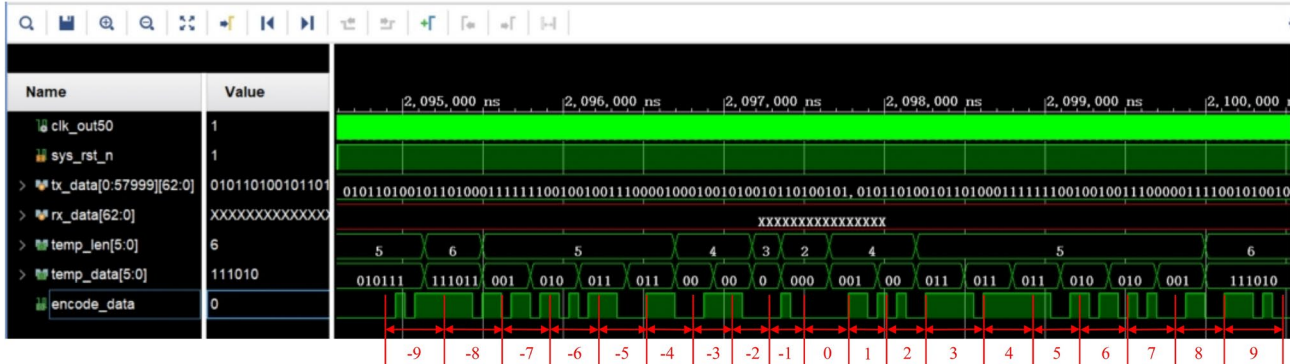


(b)  Simulation Local Magnified View

**Fig. 9**.  Simulation diagram of bitstream output of coding table.



(a)  Simulation Global Map



(b)  Simulation Local Magnified View

**Fig. 10**.  Output diagram of coded bitstream of 5th logging file.

increasing trend, the azimuth angle gradually increases, and the toolface angle undergoes significant fluctuations as the drilling tool moves. After applying residual prediction between frames using FPGA, the predicted residuals are obtained as depicted in Fig. 11c. In Fig. 11c, the calculation results for the inclination and azimuth angles are generally small and relatively stable, with only a few exceptions of larger residual values. The prediction residuals of the toolface angle still display some fluctuations, preserving the large fluctuation characteristic of the original data. However, compared to the initial fluctuation trend, there is a noticeable degree of reduction. Adaptive-range Huffman coding is then applied to the predicted residuals, and the FPGA outputs the encoded bitstream to the slave computer. Python 3.11.0 software is utilized on the slave computer for decoding verification, and the transmitted attitude data is employed for trajectory prediction, as illustrated in Fig. 11d. By comparing the predicted trajectory (d) with the attitude trajectory (a), the results are found to be completely consistent, indicating accurate transmission of the residual prediction coding. It should be noticed that there is no noise interference in the channel during the ground test, which prevents the generation of erroneous coding, resulting in a bit error rate of 0.
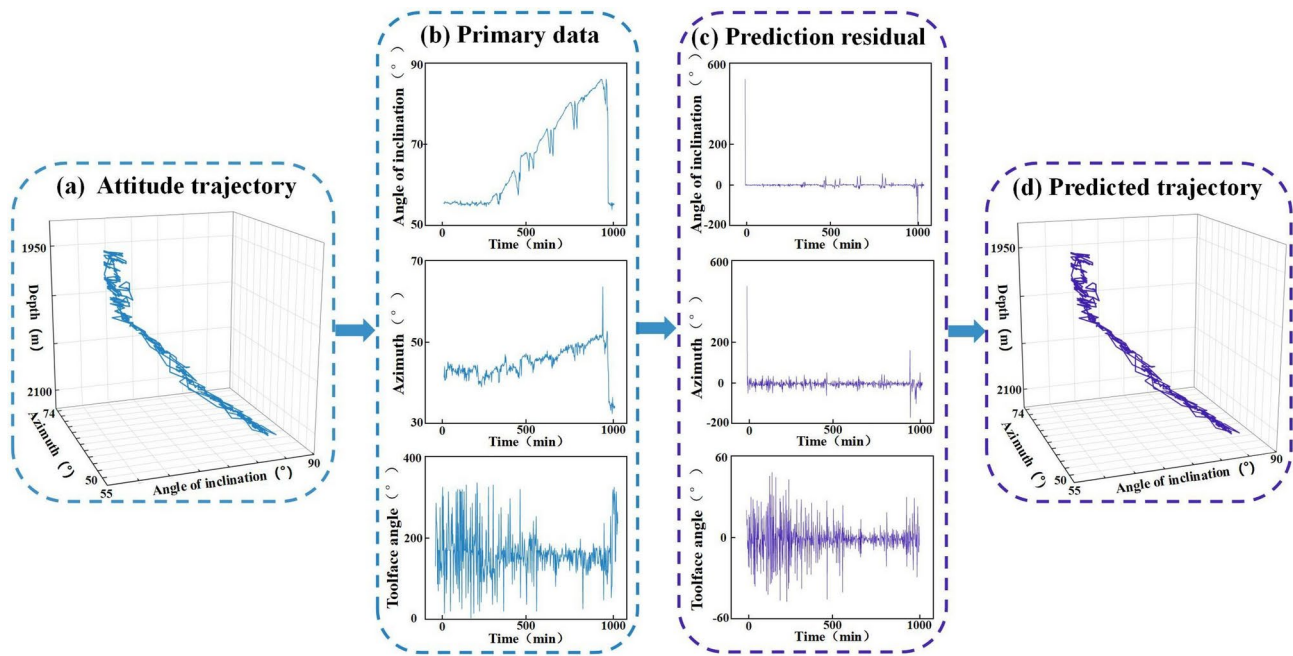
**Fig. 11**. The process of real attitude data from compression to restoration.

| Number | Filename | Attitude | $M$ | $N_1$ | $N_2$ | $C_{R1}$ | $C_{R2}$ | $\overline{C_R}$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 5th | INC | 6432 | 2375 | 2470 | 2.71 | 2.60 | 2.66 |
| | | AZI | 6432 | 2734 | 2887 | 2.35 | 2.23 | 2.29 |
| | | TF | 3752 | 3270 | 3361 | 1.15 | 1.12 | 1.13 |
| 2 | 6th | INC | 6108 | 2282 | 2114 | 2.68 | 2.89 | 2.78 |
| | | AZI | 6108 | 2863 | 2747 | 2.13 | 2.22 | 2.18 |
| | | TF | 3563 | 2575 | 2554 | 1.38 | 1.40 | 1.39 |
| 3 | 7th | INC | 251268 | 64110 | 63947 | 3.92 | 3.93 | 3.92 |
| | | AZI | 251268 | 65232 | 65277 | 3.85 | 3.85 | 3.85 |
| | | TF | 146573 | 105673 | 111025 | 1.39 | 1.32 | 1.35 |
| 4 | 8th | INC | 708660 | 176415 | 177597 | 4.02 | 3.99 | 4.00 |
| | | AZI | 708660 | 178164 | 178268 | 3.98 | 3.98 | 3.98 |
| | | TF | 413385 | 279309 | 288036 | 1.48 | 1.44 | 1.46 |

**Table 5**. Normal Distribution Curve Fitting Parameters.

Using the probability coding tables $T_1$ and $T_2$, coding and compression tests were performed on four well-logging files (5th, 6th, 7th, and 8th). The algorithm's transmission performance is summarized in Table 5. In the table, $N_1$ represents the number of bits compressed according to the coding table $T_1$, and $C_{R1}$ denotes the compression ratio corresponding to coding table $T_1$. $N_2$ represents the number of bits compressed according to the coding table $T_2$, and $C_{R2}$ indicates the compression ratio corresponding to coding table $T_2$. $\overline{C_R}$ is the average of the compression ratios for the two models.

The table illustrated that:

(1) The experiments conducted using Huffman coding tables $T_1$ and $T_2$, which were trained based on two different logging files, resulted in similar compression ratios $C_{R1}$ and $C_{R2}$. This indicates that Huffman coding tables constructed on the basis of small-range residuals do not rely on specific well logging files and exhibit compatibility with the predicted residual characters.

(2) The maximum average compression ratios $\overline{C_R}$ for the inclinations, azimuth angles, and toolface angles reach 4.00 times, 3.98 times, and 1.46 times, respectively, demonstrating the algorithm's ability to compress the attitude data effectively.

(3) The data compression ratios of 7th and 8th are notably higher than those of 5th and 6th. This indicates that the compression effect becomes more evident as the number of data frames contained in the well logging file increases. The experimental results show that the proposed algorithm is an effective and reliable compression method for drilling tool attitude data.
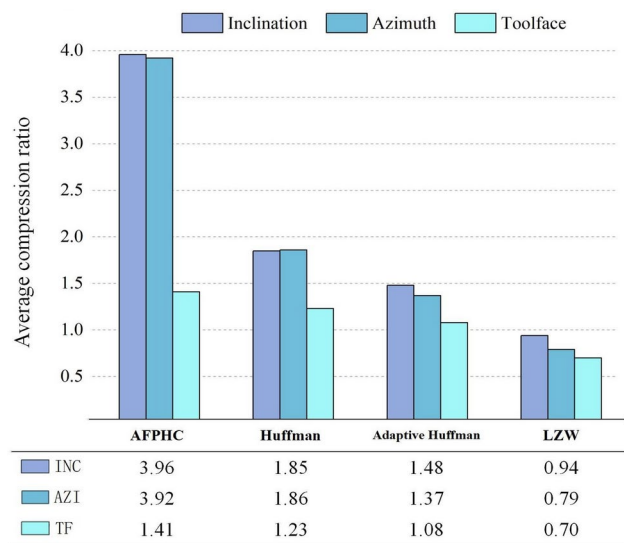
**Fig. 12**. Comparison of compression ratio.

## Discussion

Huffman, adaptive Huffman, LZW, and other lossless compression algorithms are implemented in Python 3.11.0 as baseline methods to comparatively assess the compression efficiency of the proposed algorithm. Two well-logging files, 7th and 8th, from $Group_2$, were used as the test set. Based on the principles of these algorithms, the two well-logging files were compressed and encoded, and the average compression ratios were calculated. The comparison results of different lossless compression algorithms are shown in Fig. 12. From Figure, it can be observed that the algorithm proposed in this paper achieves a higher average compression ratio $\overline{C_R}$ compared to other lossless algorithms, with average compression ratios of 3.96 times, 3.92 times, and 1.41 times for the good inclination, azimuth angle, and toolface angle, respectively. This confirms that the residual-based prediction algorithm for attitude data compression proposed in this paper achieves a higher compression ratio than those of other tested lossless compression algorithms.

While the proposed residual-based prediction algorithm and Adaptive Frame Prediction Huffman Coding (AFPHC) technique demonstrate significant improvements in attitude data compression, there are certain limitations and potential directions for future research: (1) Certain limitations: This method requires further optimization of timing paths and enhanced clock domain constraint management. Through precise timing budgeting and FPGA logic resource allocation, energy consumption and hardware overhead can be reduced, achieving optimal configuration and utilization of hardware resources. (2) Potential directions: The method's proven efficacy supports its practical adoption. Immediate extensions include aerospace applications, particularly attitude data compression, while exploring cross-domain adaptability for industrial automation and real-time embedded systems. Future developments will focus on enhancing compression performance through the integration of AI algorithms and cloud computing.

While the residual prediction in AFPHC introduces moderate computational complexity, this is effectively mitigated by the parallel architecture of FPGAs. In contrast, deep learning methods present prohibitively complex implementations for embedded systems. The FPGA implementation significantly reduces latency and achieves faster processing than software-based solutions. AFPHC excels in real-time, high-accuracy compression of MWD attitude data, combining FPGA efficiency with algorithmic innovation. It outperforms existing methods in both compression ratio and hardware adaptability, making it an ideal solution for resource-constrained downhole environments.

## Conclusion

This paper proposes a residual-based prediction algorithm for attitude data compression, solving the problems of large data volumes and high accuracy requirements in attitude data transmission. This method is not restricted by different downhole working conditions or drilling tools, demonstrating universal applicability. Based on this algorithm, an attitude data compression system was designed and implemented on an FPGA hardware platform. The algorithm is applied to the measurement data transmission system, where frame grouping is used to enhance the robustness of the system, and residual prediction between frames is employed to decrease the amount of transmitted data. By improving traditional coding methods, an adaptive-range Huffman coding method is established to reduce the bitstream of attitude data transmission, thereby enhancing both the transmission efficiency and accuracy of attitude data. Compared with other commonly used lossless compression methods, the compression ratio of angle of inclination, azimuth and toolface angle of this algorithm is up to 4.02 times, 3.98 times and 1.48 times, which proves that this algorithm has a very good compression effect.

## Data availability

The datasets used and/or analyzed during the current study available from the corresponding author on reasonable request.

## References

1. Huang, W., Wang, G. & Gao, D. A method for predicting the build-up rate of 'push-the-bit' rotary steering system. *Natl. Gas Ind. B* **8**, 622–627. https://doi.org/10.1016/j.ngib.2021.11.010 (2021).
2. Li, X. et al. A simultaneous wireless power and data transfer method utilizing a novel coupler design for rotary steerable systems. *IEEE Trans. Power Electron.* **39**, 11824–11833. https://doi.org/10.1109/TPEL.2024.3409358 (2024).
3. Dai, M., Zhang, C., Pan, X., Yang, Y. & Li, Z. A novel attitude measurement while drilling system based on single-axis fiber optic gyroscope. *IEEE Trans. Instrum. Meas.* **71**, 1–11. https://doi.org/10.1109/TIM.2021.3133328 (2022).
4. Qu, J., Xue, Q. & Lu, J. Analysis on the change characteristics of waveform during the transmission of continuous wave mud pulse signal. *Alex. Eng. J.* **80**, 594–608. https://doi.org/10.1016/j.aej.2023.09.005 (2023).
5. Jia, M. et al. Channel modelling and characterization for mud pulse telemetry. *AEU-Int. J. Electron. C.* **165**, 154654. https://doi.org/10.1016/j.aeue.2023.154654 (2023).
6. Ahmadpour, S.-S., Avval, D., Darbandi, M. & Nima Jafari, A., Navimipourand Noor Ul. A new quantum-enhanced approach to ai-driven medical imaging system. *Cluster Comput.* **28**, https://doi.org/10.1007/s10586-024-04852-2 (2024).
7. Tang, H., Liu, H., Xiao, W. & Sebe, N. When dictionary learning meets deep learning: Deep dictionary learning and coding network for image recognition with limited data. *IEEE Trans. Neural Netw. Learn. Syst.* **32**, 2129–2141. https://doi.org/10.1109/TNNLS.2020.2997289 (2021).
8. Li, C. & Xu, Z. A review of communication technologies in mud pulse telemetry systems. *Electronics* **12**, 3930. https://doi.org/10.3390/electronics12183930 (2023).
9. Ain, N. U. et al. Secure quantum-based adder design for protecting machine learning systems against side-channel attacks. *Appl. Soft Comput.* **169**, 112554. https://doi.org/10.1016/j.asoc.2024.112554 (2025).
10. Ahmadpour, S. S., Navimipour, N. J., Ain, N. U., Kerestecioglu, F. & Yalcin, S. Design and implementation of a nano-scale high-speed multiplier for signal processing applications. *Nano Commun. Netw.* **41**, 100523. https://doi.org/10.1016/j.nancom.2024.100523 (2024).
11. B. Valentin, M. *et al.* In-pixel ai for lossy data compression at source for x-ray detectors. *Nuclear Instruments and Methods in Physics Research. Section A, Accelerators, Spectrometers, Detectors and Associated Equipment* **1057**, 168665, https://doi.org/10.1016/j.nima.2023.168665 (2023).
12. Dai, S. et al. An efficient lossless compression method for periodic signals based on adaptive dictionary predictive coding. *Appl. Sci.* **10**, 4918. https://doi.org/10.3390/app10144918 (2020).
13. Ahmadpour, S.-S., Navimipour, N. J., Mosleh, M., Bahar, A. N. & Yalçin, S. A nano-scale n-bit ripple carry adder using an optimized xor gate and quantum-dots technology with diminished cells and power dissipation. *Nano Commun. Netw.* **36**, 100442. https://doi.org/10.1016/j.nancom.2023.100442 (2023).
14. Song, S., Zhao, X., Fu, H., Lian, T. & Zhang, Z. Improvement and application of lossy compression algorithm for lwd data. *Drill. Prod. Technol.* **46**, 169–176. https://doi.org/10.3969/J.ISSN.1006-768X.2023.06.27 (2023).
15. Hao, X., Gao, G., Tan, H. & Yang, C. Downhole compression algorithm for remote detection acoustic logging data based on adaptive differential pulse code modulation. *Petrol. Drill. Tech.* **52**, 148–155. https://doi.org/10.11911/syztjs.2024078 (2024).
16. Yan, Z. & Zhang, J.-T. Data compression technique of downhole video image. *J. Xi'an Shiyou Univ.* **22**, 94–97 (2007).
17. Starosolski, R. Hybrid adaptive lossless image compression based on discrete wavelet transform. *Entropy* **22**, 751. https://doi.org/10.3390/e22070751 (2020).
18. Ling, K. et al. Lossy compression of statistical data using quantum annealer. *J. Electr. Meas. Instrum.* **32**, 111–118. https://doi.org/10.13382/j.jemi.2018.10.016 (2018).
19. Zhang, W. et al. Low-cost quantized compressed sensing and transmission method for ultrasonic imaging logging. *IEEE Trans. Geosci. Remote Sens.* **62**, 1–13. https://doi.org/10.1109/TGRS.2024.3430500 (2024).
20. Weinzierl, W., Mohnke, O., Kirschbaum, L., Coman, R. & Thern, H. Comparison of pca and autoencoder compression for telemetry of logging-while-drilling nmr measurements. *SPWLA 65th Annual Logging Symposium* SPWLA–2024–0088, https://doi.org/10.30632/SPWLA-2024-0088 (2024).
21. Boram, Y., Nga, T. T. N., Chia, C. C. & Ermal, R. Lossy compression of statistical data using quantum annealer. *Sci. Rep.* **12**, 3814. https://doi.org/10.1038/s41598-022-07539-z (2022).
22. Tong, L.-Y., Lin, J.-B., Deng, Y.-Y. & Ji, K.-F. Lossless compression method for the magnetic and helioseismic imager (mhi) payload. *Res. Astron. Astrophys.* **24**, 8. https://doi.org/10.1088/1674-4527/ad2cd4 (2024).
23. Ketshabetswe, L. K., Zungeru, A. M., Lebekwe, C. K. & Mtengi, B. A compression-based routing strategy for energy saving in wireless sensor networks. *Results Eng.* 102616, https://doi.org/10.1016/j.rineng.2024.102616 (2024).
24. Palunčić, F. & Maharaj, B. T. Quasi-enumerative coding of balanced run-length limited codes. *IEEE Access* **12**, 39375–39389. https://doi.org/10.1109/ACCESS.2024.3376476 (2024).
25. Qu, S. *et al.* Coherent fading suppression of distributed acoustic sensing based on time-frequency domain linkage algorithm. *Meas. Sci. Technol.* 125114, https://doi.org/10.1088/1361-6501/ad7973 (2024).
26. Atteveldt, W., van der Velden, M. & Boukes, M. The validity of sentiment analysis: Comparing manual annotation, crowd-coding, dictionary approaches, and machine learning algorithms. *Commun. Methods Meas.* **15**, 121–140. https://doi.org/10.1080/19312458.2020.1869198 (2021).
27. Su, Y., Cao, F. & Su, T. Research and implementation of acoustic logging data compression algorithm. *Comput. Meas. Control* **23**, 4114–4116+4120. https://doi.org/10.16526/j.cnki.11-4762/tp.2015.12.058 (2015).
28. Ting, L. & Xiaofeng, S. Mud channel data compression and error correction coding research. *Electr. Meas. Technol.* **39**, 133–135+144. https://doi.org/10.19651/j.cnki.emt.2016.04.031 (2016).
29. Song, S., Lian, T., Liu, W., Luo, M. & Wu, A. A lossless compression method for logging data while drilling. *Syst. Sci. Control Eng.* **9**, 689–703. https://doi.org/10.1080/21642583.2021.1981478 (2021).
30. Xie, R. & Hai, B. Lossless compression algorithm based on hybrid coding of adaptive huffman and golomb-rice for wsn. *Comput. Eng.* **42**, 86–93. https://doi.org/10.3969/j.issn.1000-3428.2016.07.015 (2016).
31. Zhang, J., Liu, Y., Cao, J. & Yang, T. Remediation of lwd data lag with hybrid real-time data using self-attention-based encoder-decoder model. *Geoenergy Sci. Eng.* **244**, 213461. https://doi.org/10.1016/j.geoen.2024.213461 (2025).
32. Sun, W., Li, H., Li, X. & Ni, W. Gpmg spin echo data compression method based on joint coding for nmr-lwd. *J. Electronic Meas. Inst.* **32**, 135–141. https://doi.org/10.13382/j.jemi.2018.01.018 (2018).
33. Collewet, G., Musse, M., El Hajj, C. & Moussaoui, S. Multi-exponential mri t2 maps: A tool to classify and characterize fruit tissues. *Magn. Reson. Imaging* **87**, 119–132. https://doi.org/10.1016/j.mri.2021.11.018 (2022).

34. Chen, J., Gao, H., Su, Z. & Fan, J. A data lossless compression method based on deep learning and application in storing of large well logging data. *Electr. Meas. Technol.* **44**, 87–93. https://doi.org/10.19651/j.cnki.emt.2105763 (2021).
35. Song, S., Zhao, X., Zhang, Z. & Luo, M. A data compression method for wellbore stability monitoring based on deep autoencoder. *Sensors* **24**, 4006. https://doi.org/10.3390/s24124006 (2024).
36. Liu, Y. Design and implementation of while drilling data compression software based on mfc. *Inf. Technol. Inf.* **09**, 118–120. https://doi.org/10.3969/j.issn.1672-9528.2020.09.037 (2020).
37. Aamir, M., Giasin, K., Tolouei-Rad, M. & Vafadar, A. A review: drilling performance and hole quality of aluminium alloys for aerospace applications. *J. Market. Res.* **9**, 12484–12500. https://doi.org/10.1016/j.jmrt.2020.09.003 (2020).
38. Munaf, S., Bharathi, A. & Jayanthi, A. N. Fpga-based low-light image enhancement using retinex algorithm and coarse-grained reconfigurable architecture. *Sci. Rep.* **14**, 28770. https://doi.org/10.1038/s41598-024-80339-9 (2024).
39. He, X., Wang, R., Wu, J. & Li, W. Nature of power electronics and integration of power conversion with communication for talkative power. *Nat. Commun.* **11**, 2479. https://doi.org/10.1038/s41467-020-16262-0 (2020).
40. Ahmadpour, S.-S. et al. A new energy-efficient design for quantum-based multiplier for nano-scale devices in internet of things. *Comput. Electr. Eng.* **117**, 109263. https://doi.org/10.1016/j.compeleceng.2024.109263 (2024).
41. Patil, S., Banerjee, S. & Tallur, S. Smart structural health monitoring (shm) system for on-board localization of defects in pipes using torsional ultrasonic guided waves. *Sci. Rep.* **14**, 24455. https://doi.org/10.1038/s41598-024-76236-w (2024).
42. Barber, B. et al. A real-time, scalable, fast and resource-efficient decoder for a quantum computer. *Nat. Electr.* **8**, 84–91. https://doi.org/10.1038/s41928-024-01319-5 (2025).
43. Navimipour, N. J., Ahmadpour, S.-S. & Yalcin, S. A nano-scale arithmetic and logic unit using a reversible logic and quantum-dots. *J. Supercomput.* **80**, 395–412. https://doi.org/10.1007/s11227-023-05491-x (2019).
44. Liu, T. et al. High-ratio lossy compression: Exploring the autoencoder to compress scientific data. *IEEE Trans. Big Data* **9**, 22–36. https://doi.org/10.1109/TBDATA.2021.3066151 (2023).
45. Kassa, S. et al. A cost- and energy-efficient sram design based on a new 5 i-p majority gate in qca nanotechnology. *Mater. Sci. Eng., B* **302**, 117249. https://doi.org/10.1016/j.mseb.2024.117249 (2024).
46. Xilinx. Vivado 2023.2. https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/vivado-design-tools/2023-2.html. Accessed: 2023-10-19 (2023).
47. The Qt Company. Qt Creator 12.0.2. https://download.qt.io/official_releases/qtcreator/12.0/12.0.2/. Accessed: 2024-02-06 (2024).
48. Python Software Foundation. Python 3.11.0. https://www.python.org/downloads/release/python-3110/. Accessed: 2022-10-24 (2022).
49. Sun, Y. et al. Genome-wide interaction association analysis identifies interactive effects of childhood maltreatment and kynurenine pathway on depression. *Nat. Commun.* **16**, 1748. https://doi.org/10.1038/s41467-025-57066-4 (2025).
50. Dong, Y., Sun, F., Ping, Z., Ouyang, Q. & Qian, L. Dna storage: research landscape and future prospects. *Natl. Sci. Rev.* **7**, 1092–1107. https://doi.org/10.1093/nsr/nwaa007 (2020).
51. Baptiste, B., Nehal, M., Aurélien, F. & Raphael, L. Realizing the entanglement hamiltonian of a topological quantum hall system. *Nat. Commun.* **15**, 10086. https://doi.org/10.1038/s41467-024-54085-5 (2024).

## Acknowledgements

## Author contributions

Fei Li provided the idea and design for this article. Fangxing Lyu was responsible for the experimental code. Z.X. drafted the first draft of the article and the drawing of charts. Y.Y. and N.Z reviewed the revised paper. All authors read and approved the final manuscript, and all relevant authors agree to its publication.

## Declarations

## Competing interest

The authors declare no competing interests.

## Additional information

**Correspondence** and requests for materials should be addressed to F.L.

**Reprints and permissions information** is available at www.nature.com/reprints.

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.