1. Introduction
As the feature size decreases, performance of electronic devices working in intense radiation environment is severely deteriorated. Energetic particles such as protons and alpha particles affect the charge distribution in sensitive areas of devices, causing single event upset (SEU) in memory devices. Error correcting circuit is an effective method to prevent SEU from happening [1].

The basic idea of error correcting is increasing the minimum hamming distance of the code word set by inserting redundant check bits into original data bits. Therefore, errors could be detected and corrected. Among several error correcting codes, Hamming code [2] is the most widely used due to its simplicity and validity, not only because its code word is shorter than repetition code and other codes [3], but also it’s easy to encode and decode. A Hamming encoder or decoder can be serial or parallel [4], serial encoder/decoder can be implemented by software in a general-purpose processor, and it is cost saving but time-consuming. Parallel encoder/decoder is usually implemented in ASIC, however, it’s time-saving.

Considering the layout areas and power consumption of encoding and decoding circuits, a parallel error correcting circuit orienting to (7,4) Hamming code is implemented with carbon nanotube field effect transistors (CNFET) in this paper. Compared with MOSFETs, CNFETs has a series of advantages such as smaller size, lower working current, higher carrier mobility and exemption from short-channel effect [5]. CNFETs is considered as the most promising device to replace MOSFETs. At present, several models for CNFETs are available [6]. Considering the model precision and simulation time cost, Stanford model for CNFETs [7, 8] is utilized in this work.

The rest of this work is organized as follows. Section 2 is the implementation of (7, 4) Hamming encoder and decoder with modulo-2 adder. Section 3 verifies the function of the proposed circuit, and
discusses the error detecting and correcting capability of grouping method in 16-bit and 32-bit application. In section 4, performance of circuits designed respectively with CNTFETs and MOSFETs is compared in terms of layout area and power consumption. Section 5 is a brief summary.

2. Designing Error Correcting Circuit with CNTFETs

2.1. Hardware Framework

![Figure 1. Hardware framework of proposed error correcting circuit](image)

The framework of the error correcting circuit is shown in figure 1, it consists of three parts: encoder, decoder and memory. Original data is encoded before storing into the memory and decoded before reading out. In this circuit, the (7, 4) Hamming code is applied, 4-bit data is input into the encoder and then 3-bit check bits are generated and stored into the memory together with the original data bits. The generator matrix and parity matrix that we use are shown as follows.

\[
G = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1
\end{pmatrix}, \quad
H = \begin{pmatrix}
0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 & 0 & 1
\end{pmatrix}
\]

(1)

With generator matrix, all allowable code words can be calculated, as it shows in Table.1.

| Code Word | Encoder | Checker | Memory |
|-----------|---------|---------|--------|
| 0000 000  | 0101 101 | 1000 011 | 1100 110 |
| 0001 111  | 0101 010 | 1001 100 | 1101 001 |
| 0010 110  | 0110 011 | 1010 101 | 1110 000 |
| 0011 001  | 0111 100 | 1011 010 | 1111 111 |

2.2. The Modulo-2 Adder

From the generator matrix and the parity matrix shown in (1), we can derive the linear relationship between the data bits, the check bits and the syndromes. In the encoder circuit, check bits are generated by the linear combination of data bits and in the decoder circuit, syndrome is generated by the linear combination of data bits and check bits,

\[
\begin{align*}
&c_2 = d_2 + d_1 + d_0 \\
&s_2 = d_0 + c_2 + c_3 + c_0 \\
&c_3 = d_3 + d_2 + d_0 \quad s_3 = d_2 + d_3 + c_1 + c_0 \\
&c_0 = d_3 + d_2 + d_0 \quad s_0 = d_1 + d_2 + c_2 + c_0
\end{align*}
\]

(2)

From equation (2) we can see that the critical component of the encoder and the decoder is modulo-2 adder, which can be built with exclusive-or gates. In figure 2, the exclusive-or gate is implemented with CNTFETs, and the 4-bit modulo-2 adder consists of 3 exclusive-or gates, table 2 lists parameters of the Stanford model for CNTFETs.
Table 2. Parameters of CNTFETs

| Parameter | Value          |
|-----------|----------------|
| $L_g$     | 32nm           |
| $L_{dd}$  | 32nm           |
| $L_{ss}$  | 32nm           |
| $T_{ox}$  | 4nm            |
| $K_{ox}$  | 16             |
| $N_t$     | 3              |
| $(n,m)$   | (19,0)         |
| Pitch     | 20nm           |

2.3. Encoder Circuit

The structure of encoder circuit depends on the generator matrix $G$. The structure of encoder circuit is built according to equation (2) and shown in figure 3.

2.4. Decoder Circuit

The structure of decoder circuit depends on the parity check matrix $H$. According to equation (3), the decoder circuit is built and shown in figure 4. The syndrome $S$ is input into the error correcting logic to correct errors in the code word. For (7, 4) Hamming code, the error correcting logic is a 3-8 decoder.
3. Function Verification of Error Correcting Circuit

3.1. Encoder
The encoder circuit has 4 inputs \(d_3, d_2, d_1, d_0\) and 7 outputs \(d_3, d_2, d_1, d_0, c_2, c_1, c_0\). To validate its function, we run simulations in HSpice with the Stanford model for CNFETs. In the simulation, the input data varies from \(0000\) to \(1111\), and the waveform of output data is shown in figure 5.

![Figure 5. Waveform of encoder circuit](image)

Comparing the waveforms in figure 5 with the allowable code words listed in table 2, we can see that the input data is encoded correctly by the encoder implemented with CNTFET.

3.2. Decoder
The decoder circuit has 7 inputs and 7 outputs, data stored in the memory is input into the decoder and then corrected data is output. To test the error correcting capability of the decoder, several error code words are needed. In the simulation, error code words listed in table 3 is input successively into the decoder, and the waveform of the syndrome is obtained and shown in figure 6. Compared with table 3, we can see that the ordinal number of the 1-bit error \(e_1\) is equal to its syndrome, so the syndrome can be used to correct 1-bit error. However, 2-bit error \(e_7, e_8\) cannot be corrected.
Table 3. List of error code words used in the simulation and their error bits

| Error code word | Error bit | Error code word | Error bit |
|-----------------|-----------|-----------------|-----------|
| $e_1=0000100$   | 5         | $e_5=1100011$  | 2         |
| $e_2=0000110$   | 3         | $e_6=0010101$  | 1         |
| $e_3=0100111$   | 6         | $e_7=1101010$  | 4,5       |
| $e_4=0110010$   | 7         | $e_8=1100010$  | 3,6       |

Figure 6. Waveform of decoder circuit

3.3. Error correcting capability

Error correction capability of Hamming code is very limited, to correct more than 1-bit errors in 16-bit data or 32-bit data, we can apply the grouping method. The original data are equally divided into several groups, each of the group contains 4-bits of data, then we apply (7, 4) Hamming code to these groups respectively. If each group contains no more than 1-bit error, the error can be corrected, otherwise, errors can only be detected. Table 4 lists the probabilities of correctable errors in 16-bit/32-bit application.

Table 4. Probabilities of correctable errors in 16-bit/32-bit application

| Number of error bits | probability of correctable errors |
|----------------------|-----------------------------------|
|                      | 16-bit data | 32-bit data |
| 1                    | 100%        | 100%        |
| 2                    | 77.78%      | 89.09%      |
| 3                    | 41.88%      | 69.29%      |

From table 4, we can conclude that all 1-bit error can be corrected, most of the 2-bit errors can be corrected with the probability of 77.78% and 89.09% in 16-bit and 32-bit applications respectively, for 3-bit error, probability of correctable errors is vastly decreased in 16-bit application, but still acceptable in 32-bit application.

4. Performance Comparison with MOSFETs

4.1. Layout Area

Area of the layout is an important performance index for integrated circuit, smaller circuit layout area means higher integration. In this part, areas of circuits implemented with CNTFETs and 32-nm MOSFETs respectively are compared in this section.
Carriers in p-type MOSFETs and n-type MOSFETs have different mobilities. With the same drain voltage and relative gate voltage, carriers in n-type moves faster than that in p-type, therefore, p-type MOSFETs in an integrated circuit have wider channels than n-type MOSFETs. In this work, width of p-type MOSFET is twice of that of n-type. For CNTFETs, carriers in p-type and n-type CNTFETs have similar mobilities due to ballistic transport in carbon nanotubes, therefore, two types of CNTFETs have equal channel width, which means that size of CNTFETs is smaller than that of MOSFETs. Layout of single exclusive-or gate implemented with two types of FETs are compared in figure 7.

![Figure 7. Layout of exclusive-xor gate implemented with (a) CNTFETs and (b) MOSFETs](image)

Area of CNTFETs exclusive-xor gate can be calculated by parameters listed in table 2, the length of a CNTFET’s channel is $L_C=32nm$ and its width is $W_C=60nm$, which is derived from CNTFET’s pitch. The area of CNTFETs exclusive-xor gate is $A_{CNT}=16 \times L_C \times W_C=337920 nm^2$. In the MOSFETs exclusive-xor gate, the length/width ratio of n-type MOSFET is 5, so $L_M=32nm$ and $W_M=160nm$. The area of MOSFETs exclusive-xor gate is $A_{MOS}=16 \times L_M \times 12 \times W_M=337920 nm^2$. From the result, we can see that $A_{CNT}$ is about 34.4% of $A_{MOS}$, which means that CNTFETs have much higher integration than MOSFETs.

4.2. Power Consumption

Power consumption of two types of encoder is calculated and compared in this section. In the simulation, input data of the two encoders are the same as it shows in figure 5, and the power consumption is measured by the total working current provided by voltage source, since both of the power supply voltage in these two circuits are $V_{DD}=0.9V$. Waveforms of the total working current of these two encoders is shown in figure 8.

![Figure 8. Working current of encoders implemented with two types of FETs](image)

As it shows in figure 8, working current of the encoder implemented with CNTFETs is lower than MOSFETs, it shows a 56.93% decrement of average working current.
5. Conclusion
A parallel error correcting circuit based on (7, 4) Hamming code is implemented with CNFETs in this work. We choose a generator matrix and its parity check matrix to build the encoder and decoder circuits. In the simulation, 4-bit data can be encoded correctly by the encoder, and the decoder is able to correct 1-bit error. Regarding the limitations of (7, 4) Hamming code, a grouping method for 16-bit and 32-bit application is proposed and its error correction capability is analysed, the result shows that it could correct most of the 2-bit errors. At last, performances of circuits implemented respectively with CNTFETs and MOSFETs are compared, and the former shows a 34.4% decrement of layout area and a 56.9% decrement of power consumption.

Acknowledgements
This work was supported by the National Natural Science Foundation of China (Grant No. 11405270).

References
[1] Bentoutou Y IEEE Trans. Aerosp. Electron. Syst. 2012 48 648–57
[2] Hamming R W Bell System Technical Journal 1950 29 147–60
[3] Sadia A, Minahil Z and Salma Z F Int. Conf. Aero. Sci. Eng. 2013 Islamabad: 1–5
[4] Sridhar K P, Agalya R and Narmatha D International Conference on Circuit Power and Computing Technology 2014, Nagercoil: 1094–98
[5] Zhang Z and Delgado-Frias J G IEEE Trans. Nanotechnol. 2012 11 788–98
[6] Al-Shaggah A, Rjoub A and Khasawneh M. IEEE Jordan Conf. Appl. Elec. Eng. Comput. Technol. 2013 Amman: 1–6
[7] Deng J and Wong H S P IEEE Trans. Electron Devices 2007 54 3186–94.
[8] Deng J and Wong H S P IEEE Trans. Electron Devices 2007 54 3195–205