# A Lightweight block cipher based on quasigroups

**Yaohui Zhao and Yunqing Xu**[*]

Department of Mathematics, Ningbo University, Ningbo 315211, China
xuyunqing@nbu.edu.cn
*corresponding author

**Keywords:** Lightweight Block Cipher, Quasigroup, S-Box, AES

**Abstract.** The extensive deployment of tiny computing devices, such as sensors, tablets and smart phones, present a requirement for encryption systems fit for low-resource equipments. Despite implementation advances, Advanced Encryption Standard (AES) is not suitable for extremely constrained environments such as sensor networks and smart phones. In this paper, we give a new lightweight block cipher based on quasigroups. A quasigroup can be viewed as a series of S-boxes. All the S-boxes of the quasigroup used in this new cipher are optimal in linearity and differential uniformity, and all the components of these S-boxes have the highest algebraic degree. We compare the performance of this new cipher with AES by using the NIST-STS, the randomness of the new cipher is better than that of AES.

## 1. Introduction

The extensive deployment of low powered systems, such as sensors, tablets and smart phones, will be an IT landscape of this century. This provides a challenging area in the design of cryptosystems since tiny computing device's constraints of low power, low memory and limited communication ranges. Most cryptosystems such as AES and RSA were designed for desktop environments, the algorithms become a drain on battery life of low powered devices. Further, with the increase of cloud services, data being transmitted to and by these devices is growing at an exponential rate [1].

A quasigroup $(Q, *)$ is a groupoid where $Q$ is a set and $*$ is a binary operation on $Q$ such that the equations

$$a * x = b \quad \text{and} \quad y * a = b$$

are uniquely solvable for each pair of elements $a, b \in Q$. $|Q|$ is called the *order* of the quasigroup $(Q, *)$. A quasigroup of order $v$ can be viewed as a series of $v$ S-boxes.

The theory of quasigroup applications in cryptology goes through a period of rapid enough growth now. Quasigroup theory is widely used in the design of hash functions [2, 3], secret sharing systems [4], authentication of a message [5, 6], zero knowledge protocols [7], stream ciphers [8, 9], and block ciphers [10, 11], etc.

Battey and Parakh designed a quasigroup block cipher with a randomly chosen quasigroup of order 256 [10]. A quasigroup of order 256 maybe too big for low memory devices, and a random chosen quasigroup may not be optimal in linearity and differential uniformity. In this paper, we will present a new lightweight block cipher based on a carefully chosen quasigroup of order 16. The new cipher is named Quasigroup Lightweight block cipher (QLW for short). The paper is organized as follows: in Section 2 we will define two kinds of string transformations based on quasigroups, e-transformation and d-transformation. In Section 3 we present the algorithm of QLW. In Section 4 we analyse the security of QLW, include the linearity, differential uniformity, algebraic degree and randomness. Section 5 contains concluding remarks.

## 2. String Transformations based on Quasigroups

Let $Q$ be a finite set and $(Q,*)$ be a quasigroup. Let $Q^+$ be the set of all nonempty words (i.e. finite strings) formed by the elements of $Q$. The elements of $Q^+$ will be denoted by $x_1x_2\ldots x_v$, where $x_i \in Q$ $(i = 1,2,\ldots,v)$ and $v$ is a positive integer. $\forall a \in Q$, we define a mapping $E_{a,*}$: $Q^+ \to Q^+$ as follows. $\forall\ x_1x_2\ldots x_v \in Q^+$,

$$E_{a,*}(\ x_1x_2\ldots x_v) = \ y_1y_2\ldots y_v$$

where

$$\begin{cases} y_1 = a * x_1, \\ y_i = y_{i-1} * x_i, i = 2,3,\cdots,v. \end{cases}$$

The mapping $E_{a,*}$ is called an *e-transformation* of $Q^+$ based on $(Q, *)$ with leader $a$, and the graphical representation of $E_{a,*}$ is shown in Figure 1.

$$\begin{tikzcd}
 & x_1 \arrow[d] & x_2 \arrow[d] & \cdots & x_{v-1} \arrow[d] & x_v \arrow[d] \\
a \arrow[ur] & y_1 \arrow[ur] & y_2 \arrow[ur] & \cdots \arrow[ur] & y_{v-1} \arrow[ur] & y_v
\end{tikzcd}$$

Figure 1. Graphical representation of e-transformation $E_{a,*}$

We define another mapping $D_{a,*}$: $Q^+ \to Q^+$ as follows. $\forall\ x_1x_2\ldots x_v \in Q^+$,

$$D_{a,*}(\ x_1x_2\ldots x_v) = \ y_1y_2\ldots y_v$$

where

$$\begin{cases} y_1 = a * x_1, \\ y_i = x_{i-1} * x_i, i = 2,3,\cdots,v. \end{cases}$$

The mapping $D_{a,*}$ is called a *d-transformation* of $Q^+$ based on $(Q, *)$ with leader $a$, and the graphical representation is shown in Figure 2.

$$\begin{tikzcd}
a \arrow[r] & x_1 \arrow[r] \arrow[d] & x_2 \arrow[r] \arrow[d] & \cdots \arrow[r] & x_{v-1} \arrow[r] \arrow[d] & x_v \arrow[d] \\
 & y_1 & y_2 & \cdots & y_{v-1} & y_v
\end{tikzcd}$$

Figure 2. Graphical representation of d-transformation $D_{a,*}$

Let $(Q,*)$ be a quasigroup, define another binary operation "\" on $Q$ as follows:

$$x \backslash y = z \quad \Leftrightarrow \quad x * z = y.$$

It is easy to see that $(Q, \backslash)$ is also a quasigroup and $(Q, \backslash)$ is called the *132-conjugate* of $(Q,*)$.

**Theorem 1**[12] Let $Q$ be a finite set, $(Q,*)$ be a quasigroup and $(Q, \backslash)$ be the 132-conjugate of $(Q,*)$. Then $\forall\ a \in Q$ and $x_1x_2\ldots x_v \in Q^+$,

$$D_{a,\backslash}(E_{a,*}(\,x_1x_2\ldots x_v)) = \ x_1x_2\ldots x_v.$$

i.e. $D_{a,\backslash}$ is the inverse bijection of $E_{a,*}$.

Table 1 is the multiplication table of a quaisgroup $(Q,*)$, where $Q$ is the set of finite fields $F_{16}$. It is easy to check that

$$E_{3,*}\,(0123456789ABCDEF) = 62A46697B3A22BC4$$
$$D_{3,\backslash}\,(62A46697B3A22BC4) = 0123456789ABCDEF$$

Table 1. The multiplication table of a quaisgroup $(Q,*)$ of order 16.

| ∗ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | C | 1 | 5 | A | F | 4 | D | 3 | 7 | 6 | 9 | E | B | 2 | 8 | 0 |
| 1 | 7 | 6 | 9 | E | B | 2 | 8 | 0 | C | 1 | 5 | A | F | 4 | D | 3 |
| 2 | 1 | C | A | 5 | 4 | F | 3 | D | 6 | 7 | E | 9 | 2 | B | 0 | 8 |
| 3 | 6 | 7 | E | 9 | 2 | B | 0 | 8 | 1 | C | A | 5 | 4 | F | 3 | D |
| 4 | 2 | B | 0 | 8 | 6 | 7 | E | 9 | 4 | F | 3 | D | 1 | C | A | 5 |
| 5 | 4 | F | 3 | D | 1 | C | A | 5 | 2 | B | 0 | 8 | 6 | 7 | E | 9 |
| 6 | B | 2 | 8 | 0 | 7 | 6 | 9 | E | F | 4 | D | 3 | C | 1 | 5 | A |
| 7 | F | 4 | D | 3 | C | 1 | 5 | A | B | 2 | 8 | 0 | 7 | 6 | 9 | E |
| 8 | 3 | D | 4 | F | A | 5 | 1 | C | 0 | 8 | 2 | B | E | 9 | 6 | 7 |
| 9 | 0 | 8 | 2 | B | E | 9 | 6 | 7 | 3 | D | 4 | F | A | 5 | 1 | C |
| A | D | 3 | F | 4 | 5 | A | C | 1 | 8 | 0 | B | 2 | 9 | E | 7 | 6 |
| B | 8 | 0 | B | 2 | 9 | E | 7 | 6 | D | 3 | F | 4 | 5 | A | C | 1 |
| C | 9 | E | 7 | 6 | 8 | 0 | B | 2 | 5 | A | C | 1 | D | 3 | F | 4 |
| D | 5 | A | C | 1 | D | 3 | F | 4 | 9 | E | 7 | 6 | 8 | 0 | B | 2 |
| E | E | 9 | 6 | 7 | 0 | 8 | 2 | B | A | 5 | 1 | C | 3 | D | 4 | F |
| F | A | 5 | 1 | C | 3 | D | 4 | F | E | 9 | 6 | 7 | 0 | 8 | 2 | B |

## 3. Description of Quasigroup Lightweight Block Cipher

Our quasigroup lightweight block cipher (QLW for short) consist of 32 rounds. The block length is 64 bits. The key lengths of 80 and 128 bits are supported. The encryption algorithm has three parts: 1. generating round keys; 2. e-transformation layer; 3. e-xor layer. The decryption algorithm has three parts: 1. generating round keys; 2. d-xor layer; 3. d-transformation layer.

Let $Q = F_{16}$, $(Q, *)$ be the quasigroup of order 16 shown in Table 1 and $(Q, \backslash)$ be the 132-conjugate of $(Q, *)$. Let $R = F_2$ and $(R, \oplus)$ be the quasigroup with the XOR operation $\oplus$ in $F_2$. It is easy to check that the 132-conjugate of $(R, \oplus)$ is itself.

**Generating round keys:** Let $h$ = 20 or 32 (80 bits key or 128 bits key respectively) and $IK = (IK_1, IK_2,\ldots,IK_h)$, where $IK_i \in F_{16}$ $(1 \le i \le h)$. Let

$$a_i = \begin{cases} i-1 \text{ (mod 16)},\ 1 \le i \le 16,\ h+17 \le i \le 64, \\ IK_{i-16},\ 17 \le i \le h+16. \end{cases}$$

and

$$a_{h,1}a_{h,2}\cdots a_{h,64} = E_{IK_h,\backslash}(E_{IK_{h-1},\backslash}(\cdots E_{IK_1,\backslash}(a_1a_2\cdots a_{64})\cdots)).$$

Let $K_i = a_{h,2i}, 1 \le i \le 32$ be the 32 round keys (shown in Figure 3).

**e-transformation layer:** Divide the source data into 64-bit blocks. Suppose $M^{(0)}$ is a block of plain text and divide $M^{(0)}$ into 4-bit integers: $M^{(0)} = M_1^{(0)} \| M_2^{(0)} \| \cdots \| M_{16}^{(0)}$. Let

$$T_1^{(i)} T_2^{(i)} \cdots T_{16}^{(i)} = E_{K_i,*}(M_1^{(i-1)} M_2^{(i-1)} \cdots M_{16}^{(i-1)}),\ 1 \le i \le 32.$$

| $IK_i$ | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ | $a_6$ | $a_7$ | $a_8$ | $a_9$ | … $a_{64}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $IK_1$ | $a_{1,1}$ | $a_{1,2}$ | $a_{1,3}$ | $a_{1,4}$ | $a_{1,5}$ | $a_{1,6}$ | $a_{1,7}$ | $a_{1,8}$ | $a_{1,9}$ | … $a_{1,64}$ |
| $IK_2$ | $a_{2,1}$ | $a_{2,2}$ | $a_{2,3}$ | $a_{2,4}$ | $a_{2,5}$ | $a_{2,6}$ | $a_{2,7}$ | $a_{2,8}$ | $a_{2,9}$ | … $a_{2,64}$ |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ ⋮ |
| $IK_h$ | $a_{h,1}$ | $\boxed{a_{h,2}}$ | $a_{h,3}$ | $\boxed{a_{h,4}}$ | $a_{h,5}$ | $\boxed{a_{h,6}}$ | $a_{h,7}$ | $\boxed{a_{h,8}}$ | $a_{h,9}$ | … $\boxed{a_{h,64}}$ |

Figure 3. Graphical representation of generating round keys $K_i$, $i = 1,2,\ldots,32$

**e-xor layer:**

1. $T_j^{(i)} = t_{j1}^{(i)} \| t_{j2}^{(i)} \| t_{j3}^{(i)} \| t_{j4}^{(i)}, j = 1,2,\cdots,16.$
2. $m_{11}^{(i)} m_{12}^{(i)} m_{13}^{(i)} m_{14}^{(i)} \cdots m_{16,1}^{(i)} m_{16,2}^{(i)} m_{16,3}^{(i)} m_{16,4}^{(i)} = E_{b_i,\oplus}(t_{11}^{(i)} t_{12}^{(i)} t_{13}^{(i)} t_{14}^{(i)} \cdots t_{16,1}^{(i)} t_{16,2}^{(i)} t_{16,3}^{(i)} t_{16,4}^{(i)}), b_i = i \mod 2.$
3. $M_j^{(i)} = m_{j1}^{(i)} \| m_{j2}^{(i)} \| m_{j3}^{(i)} \| m_{j4}^{(i)}, j = 1,2,\cdots,16.$

The encryption algorithm of QLW is shown in Table 2.

Table 2. The encryption algorithm of QLW

```
Encryption Algorithm:
Generating round keys()
for $i$ = 1 to 32 do
    e-transformation layer (state, $K_i$)
    e-xor layer (state, $i$ mod 2)
end for
```

The result of the last round of the encryption, $M^{(32)} = M_1^{(32)} \| M_2^{(32)} \| \cdots \| M_{16}^{(32)}$ is the block of cipher text. The graphical description of the encryption of round $i$ is shown in Figure 4.

| | $M_1^{(i-1)}$ | | | | $M_2^{(i-1)}$ | | | | … | $M_{16}^{(i-1)}$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $K_i$ | $T_1^{(i)}$ | | | | $T_2^{(i)}$ | | | | … | $T_{16}^{(i)}$ | | | |
| | $t_{11}^{(i)}$ | $t_{12}^{(i)}$ | $t_{13}^{(i)}$ | $t_{14}^{(i)}$ | $t_{21}^{(i)}$ | $t_{22}^{(i)}$ | $t_{23}^{(i)}$ | $t_{24}^{(i)}$ | … | $t_{16,1}^{(i)}$ | $t_{16,2}^{(i)}$ | $t_{16,3}^{(i)}$ | $t_{16,4}^{(i)}$ |
| $b_i$ | $m_{11}^{(i)}$ | $m_{12}^{(i)}$ | $m_{13}^{(i)}$ | $m_{14}^{(i)}$ | $m_{21}^{(i)}$ | $m_{22}^{(i)}$ | $m_{23}^{(i)}$ | $m_{24}^{(i)}$ | … | $m_{16,1}^{(i)}$ | $m_{16,2}^{(i)}$ | $m_{16,3}^{(i)}$ | $m_{16,4}^{(i)}$ |
| | $M_1^{(i)}$ | | | | $M_2^{(i)}$ | | | | … | $M_{16}^{(i)}$ | | | |

Figure 4. Graphical description of encryption of round $i$, $i = 1,2,\ldots,32$

**d-xor layer:** Let $C^{(0)} = C_1^{(0)} \| C_2^{(0)} \| \cdots \| C_{16}^{(0)}$ be a block of cipher text.

1. $C_j^{(i)} = c_{j1}^{(i)} \| c_{j2}^{(i)} \| c_{j3}^{(i)} \| c_{j4}^{(i)}, j = 1,2,\cdots,16.$

2. $t_{11}^{(i)}t_{12}^{(i)}t_{13}^{(i)}t_{14}^{(i)}\cdots t_{16,1}^{(i)}t_{16,2}^{(i)}t_{16,3}^{(i)}t_{16,4}^{(i)} = D_{b_i,\oplus}(c_{11}^{(i)}c_{12}^{(i)}c_{13}^{(i)}c_{14}^{(i)}\cdots c_{16,1}^{(i)}c_{16,2}^{(i)}c_{16,3}^{(i)}c_{16,4}^{(i)})$.

3. $T_j^{(i)} = t_{j1}^{(i)} \| t_{j2}^{(i)} \| t_{j3}^{(i)} \| t_{j4}^{(i)},\ j = 1, 2, \cdots, 16$.

**d-transformation layer:**

$$C^{(i+1)} = C_1^{(i+1)}C_2^{(i+1)}\cdots C_{16}^{(i+1)} = D_{K_{32-i},\backslash}(T_1^{(i)}T_2^{(i)}\cdots T_{16}^{(i)}),\ 0 \le i \le 31.$$

The encryption algorithm is shown in Table 3. $C^{(32)} = C_1^{(32)} \| C_2^{(32)} \| \cdots \| C_{16}^{(32)}$ is the block of plain text. The graphical description of the decryption of round *i* is shown in Figure 5.

Table 3. The decryption algorithm of QLW

```
Decryption Algorithm:
Generate round keys()
for $i$ = 0 to 31 do
    d-xor layer (state, $i$ mod 2)
    d-transformation layer (state, $K_{32-i}$)
end for
```

| | $C_1^{(i)}$ | | | | $C_2^{(i)}$ | | | | … | $C_{16}^{(i)}$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $b_i$ | $c_{11}^{(i)}$ | $c_{12}^{(i)}$ | $c_{13}^{(i)}$ | $c_{14}^{(i)}$ | $c_{21}^{(i)}$ | $c_{22}^{(i)}$ | $c_{23}^{(i)}$ | $c_{24}^{(i)}$ | … | $c_{16,1}^{(i)}$ | $c_{16,2}^{(i)}$ | $c_{16,3}^{(i)}$ | $c_{16,4}^{(i)}$ |
| | $t_{11}^{(i)}$ | $t_{12}^{(i)}$ | $t_{13}^{(i)}$ | $t_{14}^{(i)}$ | $t_{21}^{(i)}$ | $t_{22}^{(i)}$ | $t_{23}^{(i)}$ | $t_{24}^{(i)}$ | … | $t_{16,1}^{(i)}$ | $t_{16,2}^{(i)}$ | $t_{16,3}^{(i)}$ | $t_{16,4}^{(i)}$ |
| $K_{32-i}$ | $T_1^{(i)}$ | | | | $T_2^{(i)}$ | | | | … | $T_{16}^{(i)}$ | | | |
| | $C_1^{(i+1)}$ | | | | $C_2^{(i+1)}$ | | | | … | $C_{16}^{(i+1)}$ | | | |

Figure 5. Graphical description of decryption of round *i*, *i* = 1,2,…,32

## 4. Security Analysis

In this section, we analyse the algebraic property of the used quasigroup shown in Table 1 and the randomness of the cipher text.

### 4.1. Linearity, Differential Uniformity and Algebraic Degree

S-boxes are widely used in block ciphers and hash functions. Usually, S-boxes are the only non-linear part in Feistel network and therefor they have to be carefully chosen to make the cipher to resist all kinds of attacks. An *n*×*n*-bit S-box can be viewed as a mapping on finite fields $F_2^n$. An invertible *n*×*n*-bit S-box can be viewed as a permutation on $F_2^n$.

Let *Q* = $F_2^n$ and (*Q*, ∗) be a quasigroup. As we know that each row of the multiplication table is a permutation on *Q*. ∀ *i* ∈ *Q*, Define a permutations on *Q* as follows:

$$S_i(x) = i * x,\quad \forall x \in Q.$$

Then $S_i(x)$ is an *n*×*n*-bit S-box. For example, the quasigroup shown in Table 1 has 16 4×4-bit S-boxes. We denote these S-boxes by $Q_0, Q_1, \ldots, Q_{15}$.

$\forall\ u, v \in F_2^n$, $u = (u_0, u_1, \cdots, u_{n-1}), v = (v_0, v_1, \cdots, v_{n-1})$, the scalar product of $u$ and $v$ can be defined as

$$\langle u, v\rangle = \sum_{i=0}^{n-1} u_i v_i .$$

Let $f$ be a Boolean function with $n$ variables ( $f : F_2^n \to F_2$ ). $\forall\ a \in F_2^n$, the *Walsh coefficient* of $f$ at $a$ is defined as

$$f^W[a] = \sum_{x\in F_2^n} (-1)^{f(x)+\langle a,x\rangle} .$$

The *linearity* of $f$ is defined as

$$\mathrm{Lin}(f) = \max_{a\in F_2^n} | f^W[a] |$$

For a given S-box of $n\times n$-bits $S$ : $F_2^n \to F_2^n$ and $\forall\ b\in F_2^n \setminus \{0\}$, the *component function* of $S$ corresponding to $b$ is defined as a Boolean function $S_b$ : $F_2^n \to F_2$

$$S_b(x) = \langle b, S(x)\rangle, \qquad \forall x \in F_2^n .$$

The *linearity* of $S$ is defined as

$$\mathrm{Lin}(S) = \max_{a\in F_2^n, b\in F_2^n \setminus\{0\}} | S_b^W[a] | .$$

The linearity of an S-box gives a measure for the resistance against linear cryptanalysis. The smaller the linearity is, the more secure the S-box is against linear attack. The smallest known linearity of a 4×4-bit S-box is 4, see [13].

Let $u = (u_0, u_1, \cdots, u_{n-1}),\ v = (v_0, v_1, \cdots, v_{n-1}) \in F_2^n$ and

$$\Delta_S(u, v) = \left|\{x \in F_2^n : S(x \oplus u) \oplus S(x) = v\}\right| .$$

Define the differential uniformity of S-box $S$ as

$$\mathrm{Diff}(S) = \max_{u\neq 0, v} \Delta_S(u, v) .$$

The differential uniformity gives a measure for the resistance of $S$ against differential cryptanalysis. Similarly, the smaller the differential uniformity is, the more secure an S-box against differential cryptanalysis. It has been shown that Diff($S$) is always even and no S-box with Diff($S$) = 2, see [13]. Therefor we have Diff($S$) ≥ 4. An bijective S-box is said to be optimal if Lin($S$) and Diff($S$) reach the minimum.

**Definition 1**[13] Let $S$ be a 4×4-bit S-box. $S$ is called to be *optimal* if it fulfills the following conditions:

(1) $S$ is a bijection;

(2) Lin($S$) = 8;

(3) Diff($S$) = 4.

Another important criterion of an S-box is the algebraic degree. A Boolean function $f$ : $F_2^n \to F_2$ can be uniquely written as a polynomial with $n$ variables, i.e., there exist coefficients $c_v \in F_2^n$ such that

$$f(x_0, x_1, \cdots, x_{n-1}) = \sum_{v\in F_2^n} c_v x_0^{v_0} x_1^{v_1} \cdots x_{n-1}^{v_{n-1}}$$

The *algebraic degree* of $f$ is the maximal weight of $v$ such that $c_v \neq 0$. Each $n$×$n$-bit S-box $S$ has $2^n - 1$ components $S_b(x) = \langle b, S(x) \rangle, b \in F_2^n \setminus \{0\}$. The *algebraic degree* of $S$ is defined as the maximal degree of its components:

$$\deg(S) = \max_{a \in F_2^n \setminus \{0\}} \deg(S_a).$$

A good S-box would have high algebraic degree.

The quasigroup $(Q, *)$ shown in Table 1 is carefully chosen by computer searches. It can be check that all the 16 S-boxes of $(Q, *)$ , $Q_0$, $Q_1$,…,$Q_{15}$, are optimal, and all the 16×15 = 240 components of these S-boxes have the highest degree, degree of 3.

## 4.2. Randomness

The National Institute of Technology-Statistical Test Suite (NIST-STS) is used to evaluate the randomness of QLW with 80 bits key. The NIST-STS package gives a *P*-value and Success/Fail status for various standardized tests. Each *P*-value is the probability that a perfect random sequence generator would have produced a sequence with less random than the one being tested [14]. Each test was given a *P*-value threshold (i.e. a significance level $\alpha$). If a *P*-value result from a test exceed the value of $\alpha$, the sequence is considered to be random, otherwise, non-random. Typically, $\alpha$ is chosen in the range [0.001, 0.01].

We compared the performance of QLW with Advanced Encryption Standard-256 (AES256) using the NIST-STS. Table 4 shows the average *P*-values (over 20 runs) for the various tests. The second and the fourth columns show the average *P*-values for all zero (0x0) and all 0xF inputs, respectively, in QLW. The third and fifth columns show the tests for AES256. The sixth column is the average *P*-value for all two inputs of QLW and the seventh column is the average P-value for all two inputs of AES256. The last column is the ratio of the *P*-values of QLW and AES256. We can notice that the *P*-values of these tests all cross 0.01, so, we can get a conclusion that the cipher text sequence is random. In addition, the proposed new block cipher, QLW, performs better than AES256.

Table 4. Average *P*-values (over 20 runs) of QLW compared with AES256

| Test | All 0x0 input QLW | All 0x00 input AES | All 0xF input QLW | All 0xFF input AES | *P*-valus for QLW | *P*-valus for AES | QLW/ AES |
|---|---|---|---|---|---|---|---|
| Block frequency | 0.51838 | 0.59109 | 0.52442 | 0.48253 | 0.52140 | 0.53681 | 0.97 |
| CS-F | 0.61461 | 0.47739 | 0.60364 | 0.36766 | 0.60913 | 0.42253 | 1.44 |
| CS-B | 0.48475 | 0.48052 | 0.43565 | 0.36949 | 0.46020 | 0.42501 | 1.08 |
| FFT | 0.27879 | 0.03377 | 0.20482 | 0.05215 | 0.24181 | 0.04296 | 5.63 |
| Frequency | 0.61017 | 0.38935 | 0.47760 | 0.29779 | 0.54389 | 0.34357 | 1.58 |
| Longest run | 0.40096 | 0.24881 | 0.50937 | 0.17118 | 0.45517 | 0.21000 | 2.16 |
| Runs | 0.43017 | 0.37347 | 0.42414 | 0.38143 | 0.42716 | 0.37745 | 1.13 |

## 5. Conclusions

In this we have presented a light weight block cipher based on a quasigroup of order 8. All the corresponding S-boxes are optimal in linearity and differential uniformity, and all the 8×15 = 120 components of these S-boxes have the highest degree, degree of 3. By using NIST-STS, we test the randomness of the new block cipher with all zero (0x0) and 0xF inputs over 20 runs and compared the reslts with that AES256 with all zero (ox00) and 0xFF inputs, the new algorithm performs better than AES256.

For future work, we intend to give more detailed analysis of the security of the new block cipher on algebraic attacks and give detailed performance.

## Acknowledgements

This research was financially supported by the National Natural Science Foundation of China under Grant No. 61373007 and Zhejiang Provincial Natural Science Foundation of China under Grant No. LY13F020039.

## References

[1] Parakh, A., Kak, S. (2009) Online data storage using implicit security. Information Sciences, 179(19): 3323-3331.

[2] Gligoroski, S., Markovski, L., Kocarev. (2009) EdonR, An Infinite Family of Cryptographic Hash functions, International Journal of Network Security, Vol. 8(3), 293–300.

[3] Snasel, V., Abraham, A., Dvorsky, J., Kromer, P., Platos, J. (2009) Hash functions based on large quasigroups. In ICCS 2009, Part I, LNCS 5544, pages 521–529, Springer-Verlag, Berlin.

[4] Cooper, J., Donovan, D., Seberry, J. (1994) Secret sharing schemes arising from latin squares. Bull. Inst. Combin. Appl., 12: 33–43.

[5] D´enes, J., Keedwell, A.D. (1992) A new authentification scheme based on latin squres. Discrete Math., 106/107: 157–165.

[6] D´enes, J., Keedwell, A.D. (2002) Some applications of non-associative algebraic systems in cryptology. P.U.M.A., 12(2):147–195.

[7] D´enes, J., D´enes, T. (2001) Non-associative algebraic system in cryptology. Protection against "meet in the middle" attack. Quasigroups Relat. Syst., 8: 7–14.

[8] Koscielny, C. (1996) A method of constructing quasigroup-based stream ciphers. Appl. Math. and Comp. Sci., 6: 109–121, 1996.

[9] Gligoroski, D., Markovski, S., Knapskong, S.J. (2008) The Stream Cipher Edon80. In: Robshaw, M. and Billet, O. (Eds.): New Stream Cipher Designs, LNCS 4986, pp. 152–169. Springer-Verlag, Berlin Heidelberg.

[10] Battey, M., Parakh, A. (2013) An Efficient Quasigroup Block Cipher, Wireless Pers Commun. 73: 63–76.

[11] Gligoroski, D., Markovski, S., Knapskog, S. J. (2008) Public key block cipher based on multivariate quadratic quasigroups, 2008. Updated and extended version of the paper presented at MATH08– Cambridge, MA, USA, March 24–26, 2008. Last revised August 2.

[12] Hu, Y., Xu, Y. (2010) Security Analysis of Cryptosystem based on Quasigroups, Proceedings of 2010 International Conference on Progress in Informatics and Computing (PIC-2010), Vol.1, 431–435.

[13] Leander, G., Poschmann, A. (2007) On the Classification of 4 bit S-boxes. In: Carlet, C., Sunar, B. (eds.) WAIFI 2007. LNCS, vol. 4547, pp. 159–176. Springer, Heidelberg.

[14] Rukhin, A., Soto, J. et al (2010). A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, NIST, Special Publication 800-22, Revision 1a.