Reprinted from JOURNAL OF COMBINATORIAL THEORY, Series A
All Rights Reserved by Academic Press, New York and London

Vol. 44, No. 2, March 1987
Printed in Belgium

Self-Dual Sequences

TUVI ETZION*

Department of Electrical Engineering Systems, University of Southern California, Los Angeles, California 90089

Communicated by the Managing Editors

Received June 5, 1986

A study is made on self-dual sequences. Some enumeration problems on the number of these sequences and the number of cycles of length k which can be produced by an n-stage shift register are investigated. Also, some full cycles with special properties are constructed from those sequences.

1. INTRODUCTION

A feedback shift-register (FSR) of span n has 2^n states corresponding to the set B^n of all binary n-tuples. The feedback function $f(x)$, $x=(x_1, x_2,..., x_n) \in B^n$, of the FSR induces a mapping $F: B^n \to B^n$ under which $xF=y$, where

$$y_i = x_{i+1} \qquad i=1,..., n-1, \quad \text{and} \quad y_n = f(x).$$

The *companion* x' of a state $x=(x_1, x_2,..., x_n)$ is defined by

$$x' = (x_1,..., x_{n-1}, x_n \oplus 1),$$

where $\oplus$ denotes modulo 2 addition.

An (n, k)-*cycle* C of an FSR of span n is a (cyclic) sequence of k digits $C=[c_1, c_2,..., c_k]$, where k is the least period of the sequence, each n consecutive digits correspond to a different state, and consecutive states in C correspond to consecutive states of the mapping F of the FSR. $[c_1, c_2,..., c_k]$ is an equivalence class where $[y_1,..., y_k]$ is *equivalent* to $[x_1,..., x_k]$ if $x_i = y_{i+p}$ for some p and all i, where subscripts are taken mod k. Two cycles C_1 and C_2 are said to be *adjacent* if they are (state) adjoint and there exists a state x on C_1 whose companion is on C_2. Two

* This research was supported in part by the Office of Naval Research under Contract N00014-84-K-0189. The author is on leave of absence from the Computer Science Department, Technion, Haifa, Israel.

0097-3165/87 $3.00

adjacent cycles C_1 and C_2, with x on C_1 and x' on C_2, are joined into a single cycle when the predecessors of x and x' are interchanged. The states x and x' are is called the *bridging states* of the join. Note that two different sets of bridging states for the same set of cycles, will produce different cycles, A Comprehensive work on shift-register cycles (or equivalently, sequences) can be found in [1].

For a sequence $S=[s_1, s_2,..., s_k]$ we define the *complement* of the sequence by $[s_1 \oplus 1, s_2 \oplus 1,..., s_k \oplus 1]$. A *self-dual sequence* (cycle) is a sequence S for which its complement is the same as S. For example $S=[0011]$ and its complement $[1100]$ are equivalent sequences. Hence $[0011]$ is a self-dual sequence. More information on the complement and self-dual sequences can be found in [2, 3].

Self-dual sequences have important roles in the parity of the number of (n, k)-cycles generated by all the shift register of span n, where a cycle is counted once for each shift-register of span n which produces it [4]. In this paper it will be shown that they also have an important role in the parity of the number of (n, k)-cycles, $\beta(n, k)$. This number is the same as the number of cycles of length k on the de Bruijn graph of order n. The known results on $\beta(n, k)$ can be found in [5].

In Section 2 of this paper we deal with some problems concerning the enumeration of self-dual sequences . We also give some new results on the parity of $\beta(n, k)$.

The join of all the cycles from a nonsingular shift-register of order n produces a full cycle (also called a de Bruijn sequence) of length 2^n. The function $f(x_1, x_2,..., x_n)$ can be considered as a logic function. It is well known [1] that $f(x_1, x_2,..., x_n)=x_1 \oplus g(x_2,..., x_n)$. Hence, the first 2^{n-1} rows of the truth table of the logic function are the complement of the last 2^{n-1} rows. The weight of the truth table is the number of Ones in the first 2^{n-1} rows. It is well known [6] that the maximum weight of the truth table of a full cycle is constructed by joining the cycles of the FSR called CCR_n which has only self-dual cycles. In Section 3 we show how to generate many de Bruijn sequences with maximum weight truth tables. A comprehensive survey on full cycles can be found [6].

2. Enumeration of Sequences

Golomb [1] defined four special registers of span n. The *pure cycling register* (PCR_n) whose feedback function is $f(x_1, x_2,..., x_n)=x_1$, the *complemented cycling register* (CCR_n) whose feedback function is $f(x_1, x_2,..., x_n)=x_1 \oplus 1$, the *pure summing register* (PSR_n) whose feedback function is $f(x_1, x_2,..., x_n)=x_1 \oplus x_2 \oplus \cdots \oplus x_n$, and the *complemented summing register* (CSR_n) whose feedback function is $f(x_1, x_2,..., x_n)=$

$x_1 \oplus x_2 \oplus \cdots \oplus x_n \oplus 1$. Denoting the number of cycles from these registers by $Z(n)$, $Z^*(n)$, $S(n)$, and $S^*(n)$, respectively he derived the following expressions:

$$Z(n) = \frac{1}{n} \sum_{d|n} \phi(d)\, 2^{n/d}$$

where $\phi(d)$ is Euler's ϕ-function, and the summation is over all divisors d of n.

$$Z^*(n) = \frac{Z(n)}{2} - \frac{1}{2n} \sum_{2d|n} \phi(2d)\, 2^{n/2d} = \frac{1}{2n} \sum_{\text{odd } d|n} \phi(d)\, 2^{n/d}$$

$$S(n) + S^*(n) = Z(n+1)$$

$$S^*(n) = Z^*(n+1).$$

All the cycles of the CCR_n are self-dual and we will derive some expressions for $S(n)$, $Z(n)$, and $Z^*(n)$ in terms of the number of self-dual sequences of length n, $\mathrm{SD}(n)$. We also give an expression for $\mathrm{SD}(n)$.

An important tool to deal with self-dual sequences is the mapping $\mathbf{D}$ which was defined by Lempel [3]. This mapping effects a two-to-one map from B^n to B^{n-1}. For a state $x = (x_0, x_1, \ldots, x_{n-1}) \in B^n$, $\mathbf{D}x = (x_0 \oplus x_1, x_1 \oplus x_2, \ldots, x_{n-2} \oplus x_{n-1}) \in B^{n-1}$, and the inverse images of x are $\mathbf{D}^{-1}x = \{(0, x_0, x_0 \oplus x_1, \ldots, \sum_{i=0}^{n-1} x_i)\ (1, 1 \oplus x_0, 1 \oplus x_0 \oplus x_1, \ldots, 1 \oplus \sum_{i=0}^{n-1} x_i)\} \subset B^{n+1}$. Note that $\mathbf{D}^{-1}x$ and $\mathbf{D}^{-1}x'$ contain two pairs of companion states. For a sequence $S = [s_0, s_1, s_2, \ldots, s_{k-1}]$, $\mathbf{D}S = [s_0 \oplus s_1, s_1 \oplus s_2, \ldots, s_{k-2} \oplus s_{k-1}, s_{k-1} \oplus s_0]$. The *weight* $W(S)$ of a sequence S is the number of Ones in S.

If $W(S)$ is even, then the inverse images of S are two complementary sequences given by

$$\mathbf{D}^{-1}S = \left\{ \left[0, s_0, s_0 \oplus s_1, \ldots, \sum_{i=0}^{k-1} s_i \right], \left[1, 1 \oplus s_0, 1 \oplus s_0 \oplus s_1, \ldots, 1 \oplus \sum_{i=0}^{k-2} s_i \right] \right\}.$$

If $W(S)$ is odd, then the inverse image of S is a self-dual sequence given by

$$\mathbf{D}^{-1}S = \left[0, s_0, s_0 \oplus s_1, \ldots, \sum_{i=0}^{k-2} s_i, 1, 1 \oplus s_0, 1 \oplus s_0 \oplus s_1, \ldots, 1 \oplus \sum_{i=0}^{k-2} s_i \right].$$

The last property is summarized in the following lemma.

LEMMA 1 [3]. *The mapping* $\mathbf{D}$ *effects as one-to-one mapping from the self-dual* $(n, 2k)$*-cycles to the* $(n-1, k)$*-cycles of odd weight.*

Applying $\mathbf{D}^{-1}$ to the cycles of linear FSR is equivalent to multiplying the characteristic polynomial of the FSR by $x+1$. By applying $\mathbf{D}^{-1}$ to the PSR_n cycles we produce the PCR_{n+1} cycles, and by applying $\mathbf{D}^{-1}$ to the CSR_n cycles we produce the CCR_{n+1} cycles. Since each PCR_n cycle of length n is an (n, n)-cycle, the PCR_n cycles contain all the (n, n)-cycles, and each self-dual sequence of length $2n$ is an $(n+1, 2n)$-cycle it follows from Lemma 1 that the number of self-dual sequences of length $2n$ is equal to the number of PCR_n cycles of length n with odd weight. It should be mentioned that the number of PCR_n cycles of length n is $(1/n) \sum_{d|n} \mu(d)\, 2^{n/d}$ [2, 7], where $\mu(d)$ is the Möbius function. In the following lemmas we find an expression for the number of self-dual cycles of length n, $\mathrm{SD}(n)$. Note that for odd n, $\mathrm{SD}(n)=0$ since the weight of a sequence S of length n is different from the weight of its complement and hence S is not self-dual.

LEMMA 2 [8]. *Let $N(n, \varepsilon)$ be the number of PCR_n cycles of length n and weight $W=\varepsilon n$, $0 \leqslant \varepsilon \leqslant 1$. Then $N(n, \varepsilon) = 1/n \sum_{d|n} \binom{d}{\varepsilon d} \mu(n/d)$, where the binomial coefficient $\binom{a}{b}$ is defined as zero if b is not an integer.*

LEMMA 3. *The number of PCR_n cycles of length n and odd weight is $1/n \sum_{d|n} \mu(n/d)\, Q(n, d)$, where $Q(n, d) = 2^{d-1}$ if n/d is odd and $Q(n, d)=0$ if n/d is even.*

Proof. The number of PCR_n cycles of length n and odd weight is $\sum_{k=0}^{\lfloor n/2 \rfloor} N(n, (2k+1)/n)$. By Lemma 2,

$$\sum_{k=0}^{\lfloor n/2 \rfloor} N(n, (2k+1)/n) = \sum_{k=0}^{\lfloor n/2 \rfloor} \frac{1}{n} \sum_{d|n} \binom{d}{(2k+1)\, d/n} \mu\left(\frac{n}{d}\right).$$

By changing the order of the summing we have

$$\sum_{k=0}^{\lfloor n/2 \rfloor} \frac{1}{n} \sum_{d|n} \binom{d}{(2k+1)\, d/n} \mu\left(\frac{n}{d}\right) = \frac{1}{n} \sum_{d|n} \mu\left(\frac{n}{d}\right) \sum_{k=0}^{\lfloor n/2 \rfloor} \binom{d}{(2k+1)\, d/n}.$$

If n/d is even the for every k, $(2k+1)\, d/n$ is not an integer and hence $\sum_{k=0}^{\lfloor n/2 \rfloor} \binom{d}{(2k+1)\, d/n} = 0$. If n/d is odd then $(2k+1)\, d/n$ is odd or not an integer and all the odd integers between 0 and d have exactly one representation as $(2k+1)\, d/n$. Hence $\sum_{k=0}^{\lfloor n/2 \rfloor} \binom{d}{(2k+1)\, d/n} = \sum_{\text{odd } k} \binom{d}{k} = 2^{d-1}$. Therefore we have

$$\frac{1}{n} \sum_{d|n} \mu\left(\frac{n}{d}\right) \sum_{k=0}^{\lfloor n/2 \rfloor} \binom{d}{(2k+1)\, d/n} = \frac{1}{n} \sum_{d|n} \mu\left(\frac{n}{d}\right) Q(n, d). \qquad \text{Q.E.D.}$$

THEOREM 1. $\mathrm{SD}(2n) = 1/n \sum_{d|n} \mu(n/d)\, Q(n, d)$.

Proof. Follows immediately from Lemmas 1 and 3. Q.E.D.

Note that $\mathrm{SD}(2n)$ is the number of self-dual cycles of period $2n$ in the de Bruijn graph of order $n+1$.

The next lemma gives two expressions of $S(n)$, $Z(n)$, and $Z^*(n)$ in terms of the number of self-dual sequences. For this purpose we need the following definition.

Let $C=[c_1,..., c_k]$ be a cycle. For a given n the complete representation $R_n(C)$ is defined by

$$R_n(C)=[x_1, x_2,..., x_{rk}],$$

where $x_{i+mk}=c_i$ and $rk=\text{l.c.m.}(k, n)$.

LEMMA 4. (a) $2S(2k-1)-Z(2k)=\sum_{d|2k}\mathrm{SD}(d)$.

(b) $Z^*(n)=\sum_{d|2n, d\nmid n}\mathrm{SD}(d)$.

Proof. (a) Applying $\mathbf{D}^{-1}$ to all the cycles of the PSR_{2k-1} we produce all the cycles of the PCR_{2k}. From each PSR_{2k-1} cycle we produce two cycles of the PCR_{2k} unless the cycle of the PSR_{2k-1} is of odd weight. In this case we produce a self-dual cycle of the PCR_{2k}. Since the PCR_{2k} contains all the cycles of length d where $d|2k$, we have that there are $\sum_{d|2k}\mathrm{SD}(d)$ self-dual cycles with the PCR_{2k} produces. Hence, $2S(2k-1)-Z(2k)=\sum_{d|2k}\mathrm{SD}(d)$.

(b) All the CCR_n cycles are self-dual with length $d=2k$, where $d|2n$. Let C be a self-dual cycle of length $d=2k$ such that $2n=dr$. Then C has the form $C=[c_1,..., c_k, c_{k+1}, c_{2k}]$, where $c_{k+i}=c_i\oplus 1$ for $1\leqslant i\leqslant k$. If $d\nmid n$ then r is odd and $R_n(C)=[x_1,..., x_n, x_{n+1},..., x_{2n}]$, where $x_{n+i}=x_i\oplus 1$ for $1\leqslant i\leqslant n$. Hence C is a cycle from the CCR_n. If $d|n$ then $R_n(C)=[x_1,..., x_n]$ and C is not a CCR_n cycle since it does not satisfy the recursion $x_{n+i}=x_i\oplus 1$. Therefore, $Z^*(n)=\sum_{d|2n, d\nmid n}\mathrm{SD}(d)$. Q.E.D.

Self-dual sequences have an important role on the parity of the number of (n, k)-cycles, $\beta(n, k)$. The following seven lemmas give important information for computing the parity of $\beta(n, k)$.

LEMMA 5. *If k is odd, the number of (n, k)-cycles with odd weight is equal to the number of (n, k)-cycles with even weight.*

Proof. Follows immediately from the fact that the weight of an (n, k)-cycle C is odd if and only if the weight of the complement of C is even. Q.E.D.

COROLLARY 5.1. *For odd k, $\beta(n, k)$ is even.*

Proof. Follows immediately from Lemma 5. Q.E.D.

By Corollary 5.1 and since only for even k, there are self-dual (n, k)-cycles it is easy to verify the following lemma.

LEMMA 6. *The parity of $\beta(n, k)$ is the same as the parity of the number of self-dual (n, k)-cycles.*

COROLLARY 6.1. *The parity of $\beta(n, k)$ is the same as the parity of the number of $(n-1, k)$-cycles with odd weight.*

Proof. Follows immediately from Lemmas 1 and 6. Q.E.D.

COROLLARY 6.2. *For odd k, $\beta(n, 2k)$ is odd if and only if $\beta(n-1, k) = 2l$ for odd l.*

Proof. Follows immediately from Lemma 5 and Corollary 6.1. Q.E.D.

LEMMA 7. *For odd k, $\beta(n, 4k) \equiv \beta(n-1, 2k)$ (mod 2).*

Proof. Since k is odd each self-dual $(n-1, 2k)$-cycle has odd weight. Also, the complement of an $(n-1, 2k)$-cycle with odd weight has odd weight. Hence, the parity of self-dual $(n-1, 2k)$-cycles is equal to the parity of $(n-1, 2k)$-cycles with odd weight. Therefore, by Lemma 6 and Corollary 6.1, $\beta(n, 4k) \equiv \beta(n-1, 2k)$ (mod 2). Q.E.D.

LEMMA 8. $\beta(n, 8k) \equiv 0$ (mod 2).

Proof. Since there is no self-dual $(n-1, 4k)$-cycle with odd weight, the number of $(n-1, 4k)$-cycles with odd weight is even. Therefore, by Corollary 6.1, $\beta(n, 8k) \equiv 0$ (mod 2). Q.E.D.

3. CONSTRUCTION OF FULL CYCLES

Fredricksen [9] made an investigation of full cycles by the weight of their truth table. The minimum weight of a truth table defining a full cycle is $Z(n) - 1$. Those sequences are generated by joining together all the PCR_n cycles. Algorithms for joining those cycles can be found found in [6, 10]. The maximum weight of truth table defining a de Bruijn sequence is $2^{n-1} - Z^*(n) + 1$. Those sequences are generated by joining together all the CCR_n cycles. The only algorithm for joining those cycles is when n is a power of 2 produces de Bruijn sequences with minimal linear complexity [11].

Let $\mathrm{FC}(\mathrm{CCR}_n)$ be the number of full cycles generated by joining the CCR_n cycles, and let $\mathrm{FC}(\mathrm{CSR}_n)$ be the number of full cycles generated by joining the CSR_n cycles.

LEMMA 9. $\mathrm{FC}(\mathrm{CCR}_{n+1}) = \mathrm{FC}(\mathrm{CSR}_n)\, 2^{Z^*(n+1)-1}$.

Proof. $\mathbf{D}$ is a one-to-one mapping from the CCR_{n+1} cycles to the CSR_n cycles. Given a set of bridging states for joining the CSR_n cycles, then since for each pair of bridging states x and x' in B^n, $\mathbf{D}^{-1}x \in B^{n+1}$ and $\mathbf{D}^{-1}x' \in B^{n+1}$ contain two pairs of companion states, we can choose one of the two pairs of states of $\mathbf{D}^{-1}x$ and $\mathbf{D}^{-1}x'$ as bridging states for the join of the CCR_{n+1} cycles. Also, since there are $S^*(n) = Z^*(n+1)$ CSR_n cycles we have $Z^*(n+1)-1$ states in the set of bridging states. Hence, $\mathrm{FC}(\mathrm{CCR}_{n+1}) \geqslant \mathrm{FC}(\mathrm{CSR}_n)\, 2^{Z^*(n+1)-1}$. In a similar way we have $\mathrm{FC}(\mathrm{CCR}_{n+1}) \leqslant \mathrm{FC}(\mathrm{CSR}_n)\, 2^{Z^*(n+1)-1}$. Therefore, $\mathrm{FC}(\mathrm{CCR}_{n+1}) = \mathrm{FC}(\mathrm{CSR}_n)\, 2^{Z^*(n+1)-1}$. Q.E.D.

The proof of Lemma 9 leads to a construction of full cycles by joining together all the CCR_{n+1} cycles. We start with a set of bridging states for a construction of a full cycle from the cycles of the CSR_n. If the states x and x' are bridging states we choose one of the members of $\mathbf{D}^{-1}x$ and it companion as bridging states for the join of all the CCR_{n+1} cycles.

An algorithm for joining together all the CSR_n cycles could be chosen in a way similar to the algorithm in [10] for joining all the PSR_n cycles together. First we will make the necessary changes from the construction in [10] for joining the PSR_n cycles to the construction for joining the CSR_n cycles (the proof of Lemmas 10, 11, and 12, and Theorem 2 are the same as in [10]).

An *extended representation* $\mathrm{E}(C)$ of a cycle C of CSR_n is given by an $(n+1)$-tuple $[x_0 x_1 \cdots x_{n-1} x_n]$, where $(x_0, x_1, \ldots, x_{n-1})$ is a state on C and $x_n = x_0 \oplus x_1 \oplus \cdots \oplus x_{n-1} \oplus 1$.

The *extended weight* $\mathrm{W_E}(C)$ of C is defined as the number of *Ones* in $\mathrm{E}(C) = [x_0 x_1 \cdots x_{n-1} x_n]$, i.e., $\mathrm{W_E}(C) = \sum_{i=0}^{n} x_i$.

The following lemma is an immediate result of the above definitions.

LEMMA 10. *For every cycle C from* CSR_n *we have* $\mathrm{W_E}(C) = 2k+1$, *for some* $0 \leqslant k \leqslant \lfloor n/2 \rfloor$, *and for each state S on C* $2k \leqslant \mathrm{W}(S) \leqslant 2k+1$.

C is called a *run-cycle* if all the *Ones* in $E(C)$ form a cyclic run.

For each cycle C of CSR_n, with $\mathrm{W_E}(C) = 2k+1 < n+1$, we define a unique *preferred state* $\mathrm{P}(C)$. For a run-cycle $\mathrm{P}(C) = (1^{2k+1}0^{n-2k-1})$; for a cycle with more than one (cyclic) run of *Ones* the preferred state is defined as follows.

Let $\mathrm{E}^*(C) = [0^r 1^t 0 b_1 \cdots b_{n-t-r-2} 10]$ be the unique extended representation of C which satisfies the following properties:

(a) $r \geqslant 0$;

(b) t is the length of the longest run of *Ones*;

(c) among all extended representations of this form, with the same maximal t, $E^*(C)$ is the largest when viewed as a number in base-2 notation.

Then, the preferred state for C is $P(C) = (0^r 1^t 0 b_1 \cdots b_{n-t-r-2} 1)$.

LEMMA 11. *Let C_1 be a nonrun-cycle from* CSR_n *and let* $P(C_1) = (0^r 1^t 0 b_1 \cdots b_{n-t_1-r-2} 1)$. *Then the states* $B = (10^r 1^t 0 b_1 \cdots b_{n-t_1-r-2})$ *and the companion of* $P(C_1)$ *are on cycle* $C_2 \neq C_1$, *with* $W_E(C_2) = W_E(C_1)$. *Furthermore, if t_2 is the length of the longest run of Ones in* $P(C_2)$ *then either* $t_2 = t_1 + 1$ *or* $t_2 = t_1$ *and* $P(C_1)$ *is greater than* $P(C_2)$ *when they are viewed as binary numbers.*

LEMMA 12. *Let* $U = (u_1, ..., u_{n-1}, 1)$ *be a state on a cycle C_1 of* CSR_n *with* $W(U) + 1 = W_E(C_1) = 2k + 1$ *for some* $k \geqslant 1$. *Then the companion U' of U is on* CSR_n *cycle* C_2 *with* $W_E(C_2) = 2k - 1$.

Lemmas 10, 11, and 12 lead to the construction of a large class of full cycles from the cycles of CSR_n. Lemma 11 suggests a way of joining all cycles with the same extended weight. For each extended weight $2k+1$, we start with the run-cycle of this weight as an initial main cycle. In each step the current main cycle is expanded by joining to it the CSR_n cycle of extended weight $2k+1$ with the longest run of *Ones*; if there are two or more cycles with the same longest run of *Ones*, join the one with the largest preferred state. Once all the CSR_n cycles of extended weight $2k+1$ are joined together into a corresponding main cycle MC_k, $0 \leqslant k \leqslant \lfloor n/2 \rfloor$, we apply Lemma 12 to joining the MC_k cycles, in order of increasing k, to form a full cycle.

We proced now to describe an algorithm for producing the $(i+n)$th bit b_{i+n} of the resulting full cycle from the following inputs:

(a) the preceding n-bit state $\beta_i = (b_i, b_{i+1}, ..., b_{i+n-1})$,

(b) the parity p_i of β_i, $p_i = b_i \oplus b_{i+1} \oplus \cdots \oplus b_{i+n-1}$, and

(c) the weight $W(\beta_i)$ of β_i.

The production of b_{i+n} from the above inputs is based on the fact that when $(x_1, ..., x_{n-1}, x_n)$ is the successor of $(x_0, x_1, ..., x_{n-1})$ then $\sum_{i=0}^{n} x_i$ is odd if and only if both states are on the same CSR_n cycle.

ALGORITHM A. For every k such that $1 \leqslant k \leqslant \lfloor n/2 \rfloor$ choose and store a bridging state $U^{(2k)}$ of the form $U^{(2k)} = (u_1^k, u_2^k, ..., u_{n-1}^k, 1)$ with $W(U^{(2k)}) = 2k$. Initially, set $\beta_0 = (0, 0, ..., 0) = 0^n$, $p_0 = 0$, and $W(\beta_0) = 0$. Given $\beta_i = (b_i, b_{i+1}, ..., b_{i+n-1})$, $p_i, w_i = W(\beta_i)$ proceed to produce $\beta_{i+1} = (b_{i+1}, ..., b_{i+n-1}, b_{i+n})$, p_{i+1}, and w_{i+1} as follows:

(A1) If $p_i \oplus b_i = 0$ go to (A3).

(A2) If $(b_{i+1},..., b_{i+n-1}, 1) = U^{(w_i - b_i + 1)}$ go to (A6); otherwise go to (A5).

(A3) If $\beta_i^* = [b_{i+1},..., b_{i+n-1} 10]$ is a run-cycle go to (A5); otherwise, find the cyclic shift $E_i^* = [0^r 1^t 0 b_s \cdots b_{n-t-r+s-3} 10]$ of β_i^* whose first n bits form a preferred state.

(A4) If $E_i^* = \beta_i^*$ go to (A6).

(A5) Set $b_{i+n} = p_i \oplus 1$, $p_{i+1} = b_i \oplus 1$, $w_{i+1} = w_i - b_i + (p_i \oplus 1)$, and stop.

(A6) Set $b_{i+n} = p_i$, $p_{i+1} = b_i$, $w_{i+1} = w_i - b_i + p_i$.

THEOREM 2. (a) *For every choice of the set of states* $\{U^{(2k)}\}_{k=1}^{\lfloor n/2 \rfloor}$ *Algorithm A produces a full cycle of length* 2^n.

(b) *Algorithm A can be used to produce*

$$\prod_{k=1}^{\lfloor n/2 \rfloor} \binom{n-1}{2k-1}$$

distinct full cycles.

(c) *The working space that Algorithm A requires to produce a full cycle is about* $n^2/2$ *bits and the work required to produce the next bit is n cyclic shifts and about the same number of n-bit comparisons.*

If the states x and x' are bridging states for the construction of the full cycle from the CSR_n cycles, then we choose the state that starts with a zero from $\mathbf{D}^{-1}x$ and its companion as bridging states for the construction of full cycle from the CCR_{n+1} cycles. If $\delta_i = (d_i, d_{i+1},..., d_{i+n})$ is a state on the full cycle then $(d_{i+1},..., d_{i+n}, 1)$ serves as a bridging state for the CCR_{n+1} cycles if and only if $d_{i+1} = 0$ and $(b_{i+1},..., b_{i+n-1}, 1)$, for $b_{i+j} = d_{i+j} \oplus d_{i+j+1}$, $1 \leqslant j \leqslant n-1$, serves as a bridging state for the CSR_n cycles. If δ_i serves as a bridging state then $d_{i+n+1} = d_i$; otherwise $d_{i+n+1} = d_i \oplus 1$. The formal steps for the production of the next bit in the full cycle of span $n+1$, d_{i+n+1}, are given in Algorithm B.

ALGORITHM B. For every k such that $1 \leqslant k \leqslant \lfloor n/2 \rfloor$ choose and store a bridging state $U^{(2k)}$ of the form $U^{(2k)} = (u_1^k, u_2^k,..., u_{n-1}^k, 1)$ with $\mathrm{W}(U^{(2k)}) = 2k$. Initially, set $\delta_0 = (0, 0,..., 0) = 0^{n+1}$, $\beta_0 = (0, 0,..., 0) = 0^n$, $p_0 = 0$, and $\mathrm{W}(\beta_0) = 0$. Given $\delta_i = (d_i, d_{i+1},..., d_{i+n})$, $\beta_i = (b_i, b_{i+1},..., b_{i+n-1})$, p_i, $w_i = \mathrm{W}(\beta_i)$ proceed to produce $\delta_{i+1} = (d_{i+1},..., d_{i+n}, d_{i+n+1})$, $\beta_{i+1} = (b_{i+1},..., b_{i+n-1}, b_{i+n})$, p_{i+1}, and w_{i+1} as follows:

(B1) If $d_{i+1} = 1$, go to (B6).

(B2) If $p_i \oplus b_i = 0$ go to (B4).

(B3) If $(b_{i+1},..., b_{i+n-1}, 1) = U^{(w_i - b_i + 1)}$ go to (B7); otherwise go to (B6).

(B4) If $\beta_i = [b_{i+1},..., b_{i+n-1} 10]$ is a run-cycle go to (B6); otherwise, find the cyclic shift $E_i^* = [0^r 1^t 0 b_s \cdots b_{n-t-r+s-3} 10]$ of β_i^* whose first n bits form a preferred state.

(B5) If $E_i^* = \beta_i^*$ go to (B7).

(B6) Set $d_{i+n+1} = d_i \oplus 1$ and go to (B8).

(B7) Set $d_{i+n+1} = d_i$.

(B8) Set $b_{i+n} = d_{i+n} \oplus d_{i+n+1}$, $p_{i+1} = p_i \oplus b_i \oplus b_{i+n}$, $w_{i+1} = w_i - b_i + b_{i+n}$.

THEOREM 3. *Algorithm* B *produces the same number of full cycles of span* $n+1$ *as the number of full cycles of span* n *which Algorithm* A *produces. The working space and the time complexity of Algorithm* B *is the same as those of Algorithm* A.

Proof. Follows directly from the discussion preceding Algorithm B, since there is only a constant number of additions in Algorithm B more then those in Algorithm A. Q.E.D.

Given all the full cycles produced from joining the cycles of the PSR_n and the CSR_n it is interesting to know whether there is an overlap between these full cycles. In the following lines it will be shown that for most n, there is no overlap. It is easy to verify that each $(n+1)$-tuple in the PSR_n (CSR_n) cycles has even (odd) weight. Joining two cycles changes the weight of two $(n+1)$-tuples by one. Hence, the number of $(n+1)$-tuples with even weight in a full cycle produced from the PSR_n (CSR_n) is $2^n - 2(S(n)-1)$ $(2(S^*(n)-1))$. For $n \geqslant 4$, $2^n - 2(S(n)-1) > 2(S^*(n)-1)$ and therefore there is no overlap between the set of full cycles produced from the PSR_n and those produced from the CSR_n.

ACKNOWLEDGMENT

The author wishes to thank the referee for his valuable suggestions and comments.

REFERENCES

1. S. W. GOLOMB, "Shift Register Sequences," Aegean Park, Laguna Hills, CA, 1982.
2. E. A. WALKER, Non-linear recursive sequences, *Canad. J. Math.* **11** (1959), 370–378.
3. A. LEMPEL, On a homomorphism of the de Bruijn graph and its applications to the design of feedback shift register, *IEEE Trans. Comput. C* **19** (1970), 1204–1209.

4. P. F. Duvall, Jr. and R. E. Kibler, On the parity of the frequency of cycle lengths of shift register sequences, *J. Combin. Theory, Ser. A* **18** (1975), 357–361.
5. P. R. Bryant and J. Christensen, The enumeration of shift register sequences, *J. Combin. Theory, Ser. A* **35** (1983), 154–172.
6. H. Fredericksen, A survey of full length nonlinear shift register algorithm, *SIAM Rev.* **24** (1982), 195–221.
7. B. Elspas, Theory of autonomous linear sequential networks, *IRE Trans. Circuit Theory CT-6* (1959), 45–60.
8. T. Etzion, An algorithm for generating shift-register cycles, *Theoret. Comput. Sci.* **44** (1986), 209–224.
9. H. Fredricksen, Classification of maximal length sequences by weight, *in* "Proceedings, Hawaii International Conference on System Science," (1969), Vol. 2.
10. T. Etzion and A. Lempel, Algorithms for the generation of full-length shift-register sequences, *IEEE Trans. Inform. Theory IT* **30** (1984), 480–484.
11. T. Etzion and A. Lempel, Construction of de Bruijn sequences of minimal complexity, *IEEE Trans. Inform. Theory IT* **30** (1984), 705–709.

Printed by the St. Catherine Press Ltd., Tempelhof 41, Bruges, Belgium