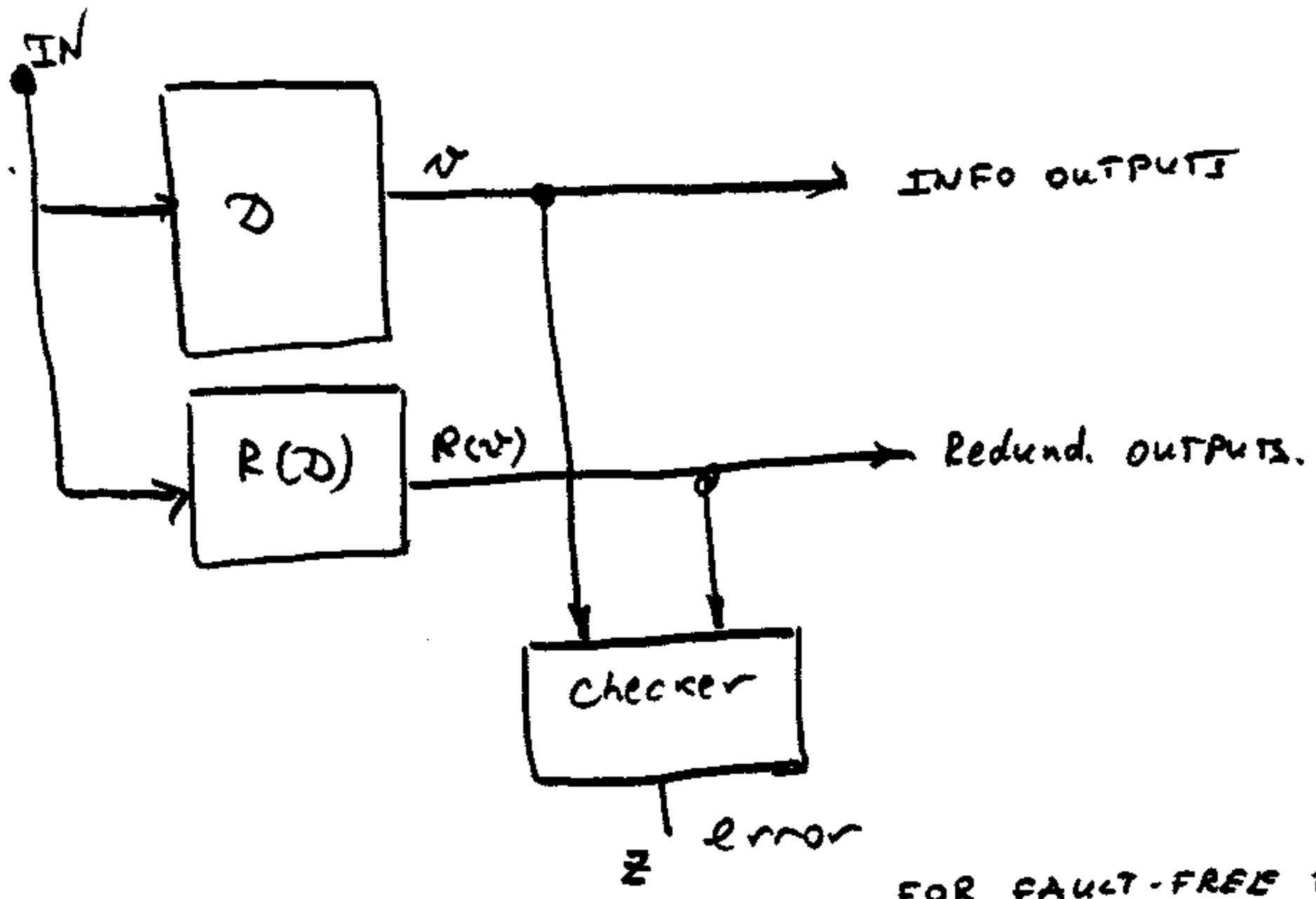
SC753. MDP 114

Self Checking Checkers

(Self-Checking Decoders for Error Detecting/Correcting Codes)



$X = (v, R(v)) \in \mathcal{X}$ - input code

$z(x)$ output

$\mathcal{Z}$ domain of $z(x)$

$\mathcal{Z}$ output code

FOR FAULT-FREE $v, R(v)$ range of $z(x)$

Consider class F of faults in the checker

Ex. F is SSFs.

Fault f in the checker

$$Z(x) \longrightarrow Z_f(x) \qquad f \in F.$$

DEF.1 Checker is fault-secure iff

for $\forall x \in X$ and $\forall f \in F$

$Z_f(x) \notin Z$

|

fault is detected by the checker

DEF2 Checker is self-testing iff

For $\forall f \in F \quad \exists x \in X : \quad Z_f(x) \notin Z$

∴ there exists at least one fault-free input which provokes a fault $f \in F$ in the checker and distorts the outputs of the checker.

DEF 3 A circuit is a checker iff

for any $x \notin X$ $\quad Z(x) \notin Z$

and $x \in X$ $\quad Z(x) \in Z$.

(This is also known as the code-disjoint property)

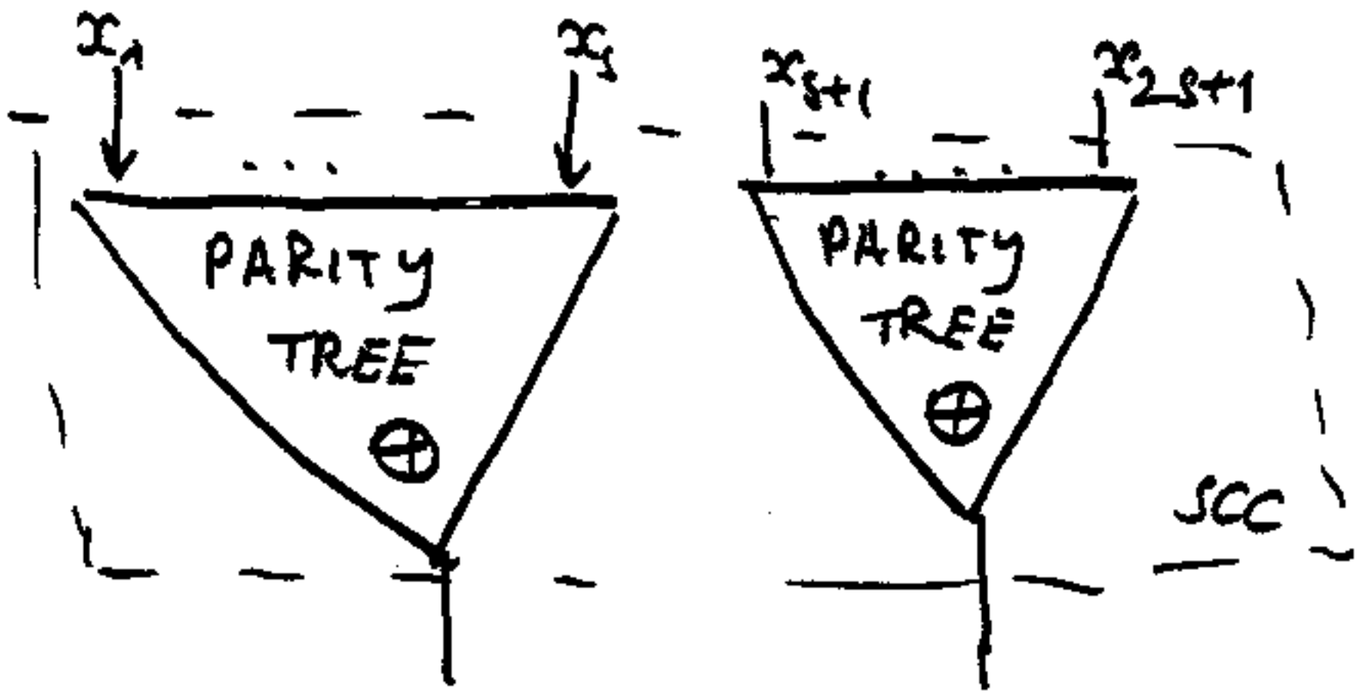
DEF 4 A checker is self-checking iff fault-secure and self-testing.

+1 A self-checking checker needs to have at least two output lines, each of which must take values 1 and 0 during normal operation.

EXAMPLES OF SELF checking checkers (SCC)

X is an odd parity code of odd length

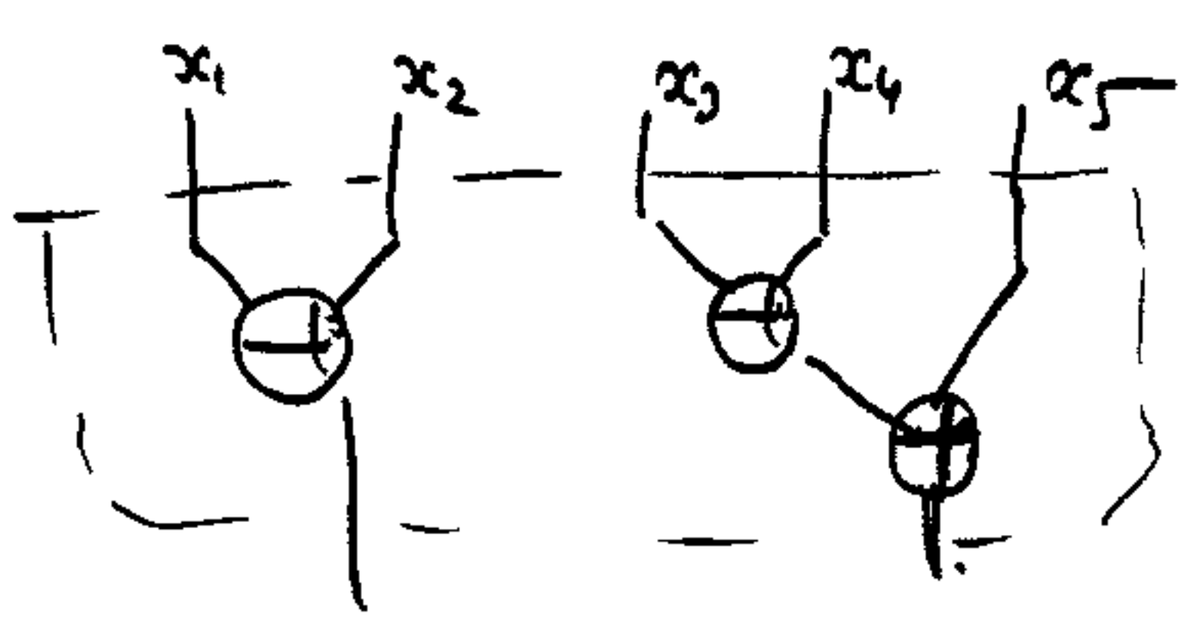
$(x_1, \ldots, x_{2s+1}) \in X$ iff $x_1 \oplus x_2 \oplus \ldots \oplus x_{2s+1} = 1$



F is a set of SSFs.

$Z = \{01, 10\}$.

$s = 2$



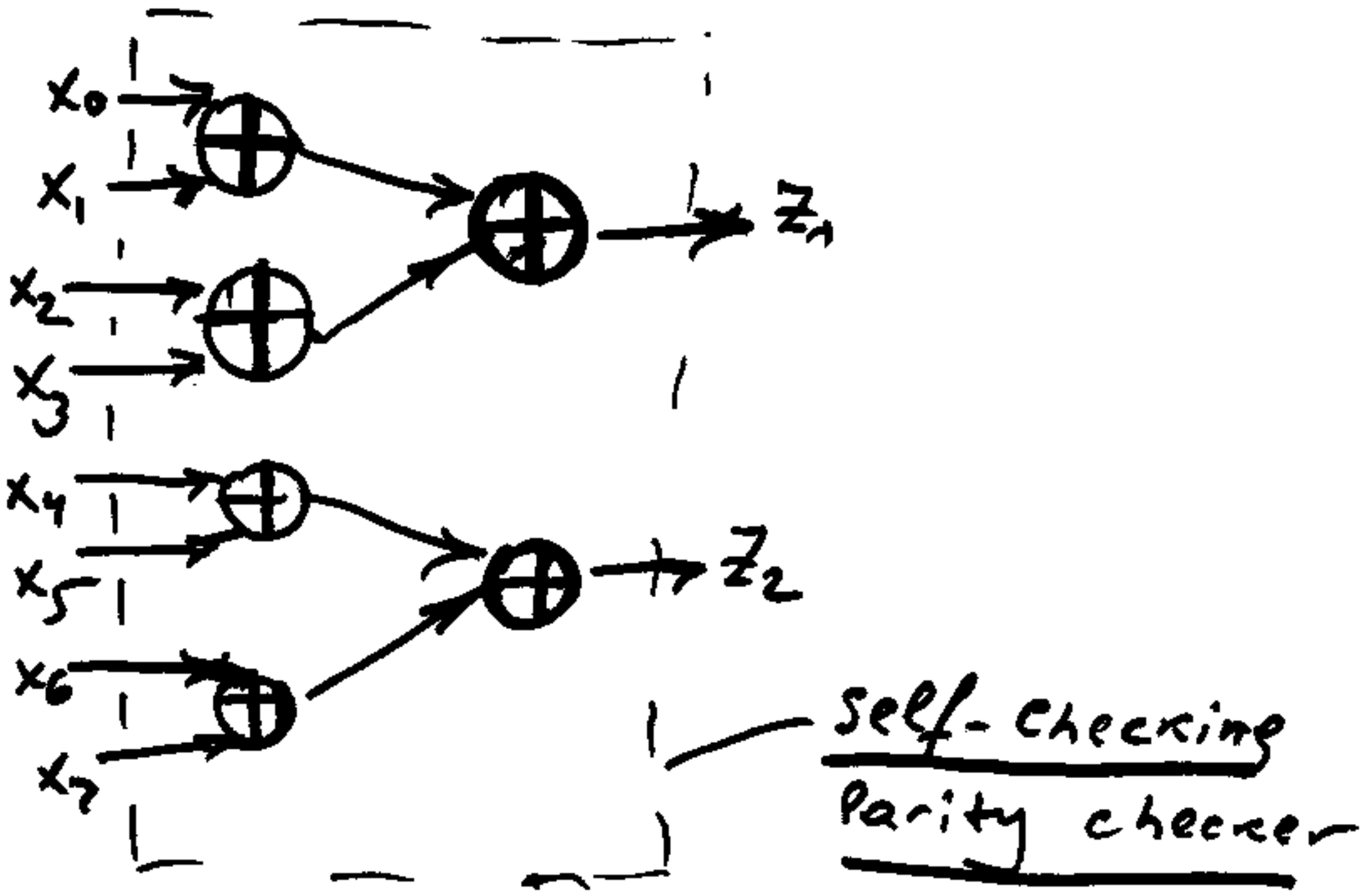
Self checking Checkers for K out of m codes and Burger codes can be found in: PRADHAN, FAULT TOLERANT COMPUTING vol 1, Ch 5.

Self-Testing Checker for

PARITY VERIFICATION

Verify $x_0 \oplus x_1 \oplus .. \oplus x_{m-1} = 0 \Leftrightarrow x \in V$

Example $m=8$



1) If $x \in V$ and no faults in the checker $\Rightarrow$
$z_1 = z_2$

2) If $x \notin V$ (odd number of ones in x) and no faults in the checker $z_1 \neq z_2$

3) If $x \in V$ and there is a SSF in the checker then $z_1 \neq z_2$.

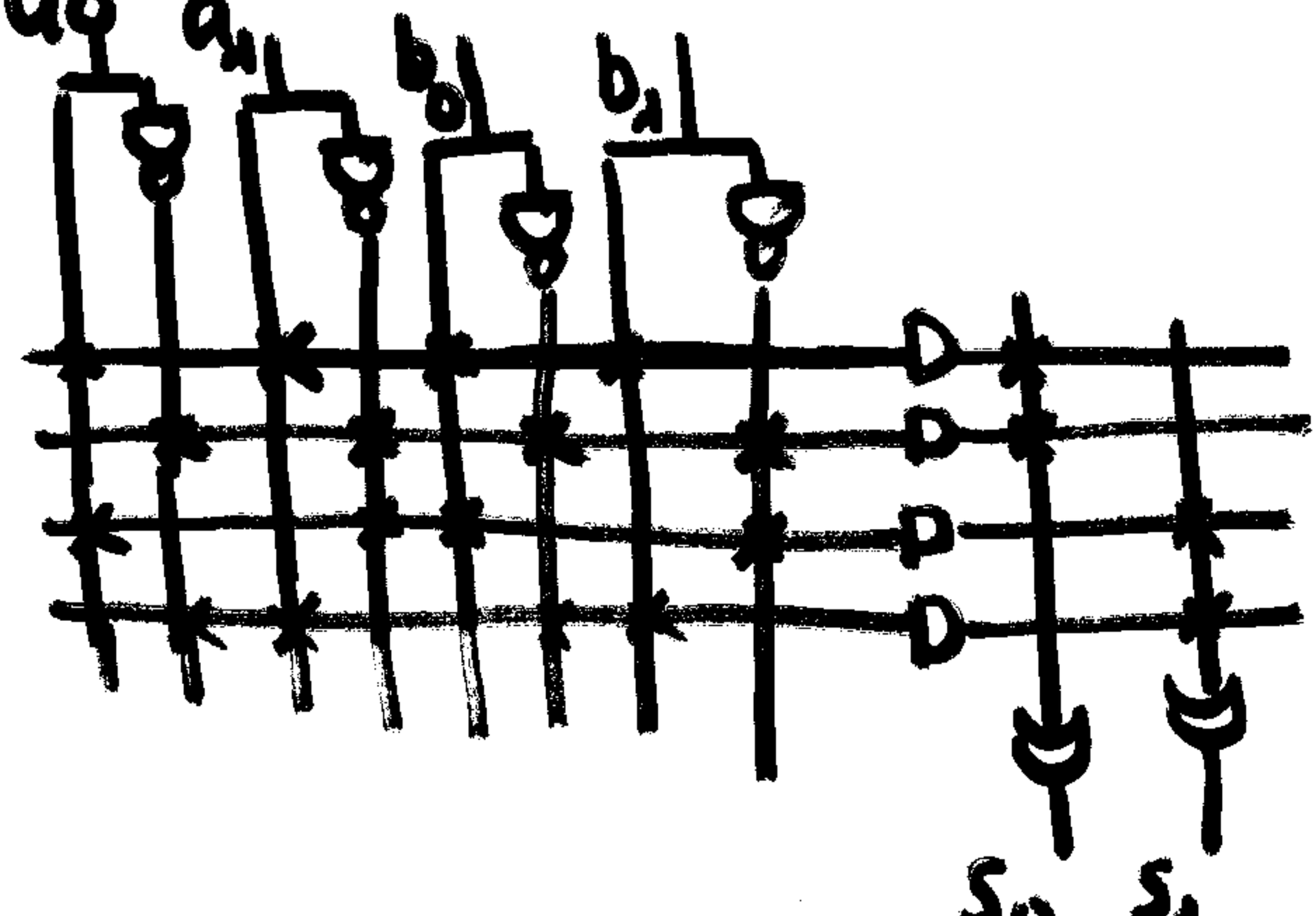
TOTALLY SELF-CHECKING MATCH DETECTOR

$K=2$

$a=(a_0,a_1)$, $b=(b_0,b_1)$

$(a,b)\in C_{In} \Leftrightarrow a=b$

$C_{out}=\{01,10\}$



$s_0=1 \quad a_0=b_0=a_1=b_1$

$s_1=1 \quad a_0=b_0 \neq a_1=b_1$

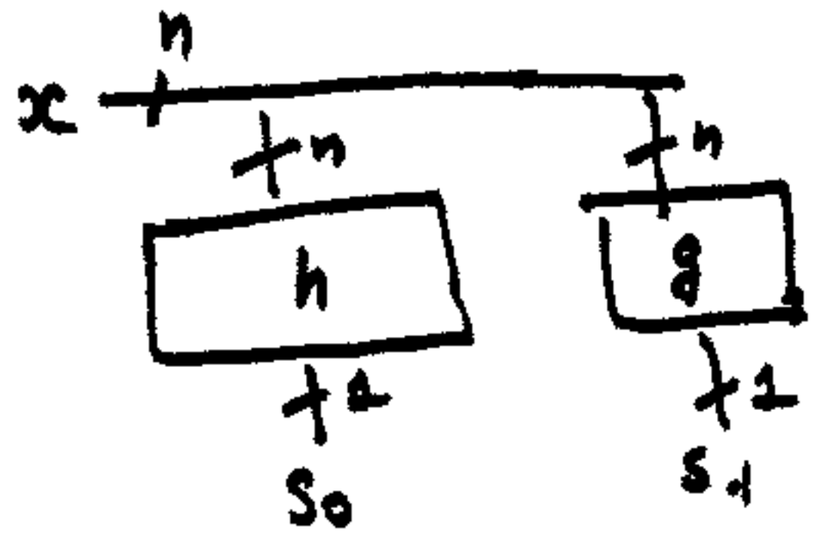
DESIGN OF TOTALLY SELF-CHECKING CHECKERS (TSCC)

Let $f(x) = 1 \Leftrightarrow x \in C_{IN}$ - INPUT code

$x \in \{0,1\}^n$

Represent f as $h(x) \oplus g(x)$

Then



is a TSCC with $C_{out} = \{01, 10\}$

iff C_{IN} is a test set for SSFs in networks implementing h and g.

DUAL-RAIL DESIGN OF TSCC with $C_{out} = \{01, 10\}$

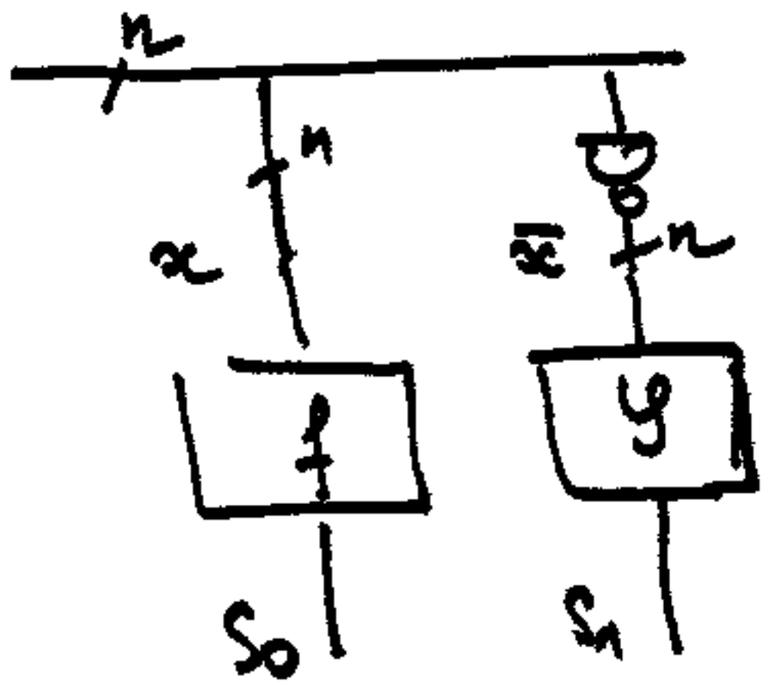
Let $f(x) = 1 \Leftrightarrow x \in C_{out}$ $\quad x \in \{0,1\}^n$, $C_{out} \subseteq \{0,1\}^n$

Denote $g(x) = \overline{f}(\overline{x})$

$\overline{x}$ - componentwise negation of x

$g(x)$ called <u>dual</u> to $f(x)$

$$g(\overline{x}) = \overline{f}(x)$$



C_{IN} is a test for SSFs in f and g

f and g have the same complexity.

This approach is better than replication (duplication) of f (duplication does <u>not</u> provide for self-testing)

If $f(x_1, x_2, \ldots, x_n) = 1 \Leftrightarrow$

$(x_1, \ldots, x_n) \in C_{IN}$ and f depends on all x_i and $\overline{x_i}$ (non-unary in all x_i)

then AND-XOR Reed-Muller implementations of f and g (g is dual to f) generates a dual-rail TSCC for C_{IN} with $C_{OUT} = \{01, 10\}$

Example 3 MR ($\kappa = 1$)

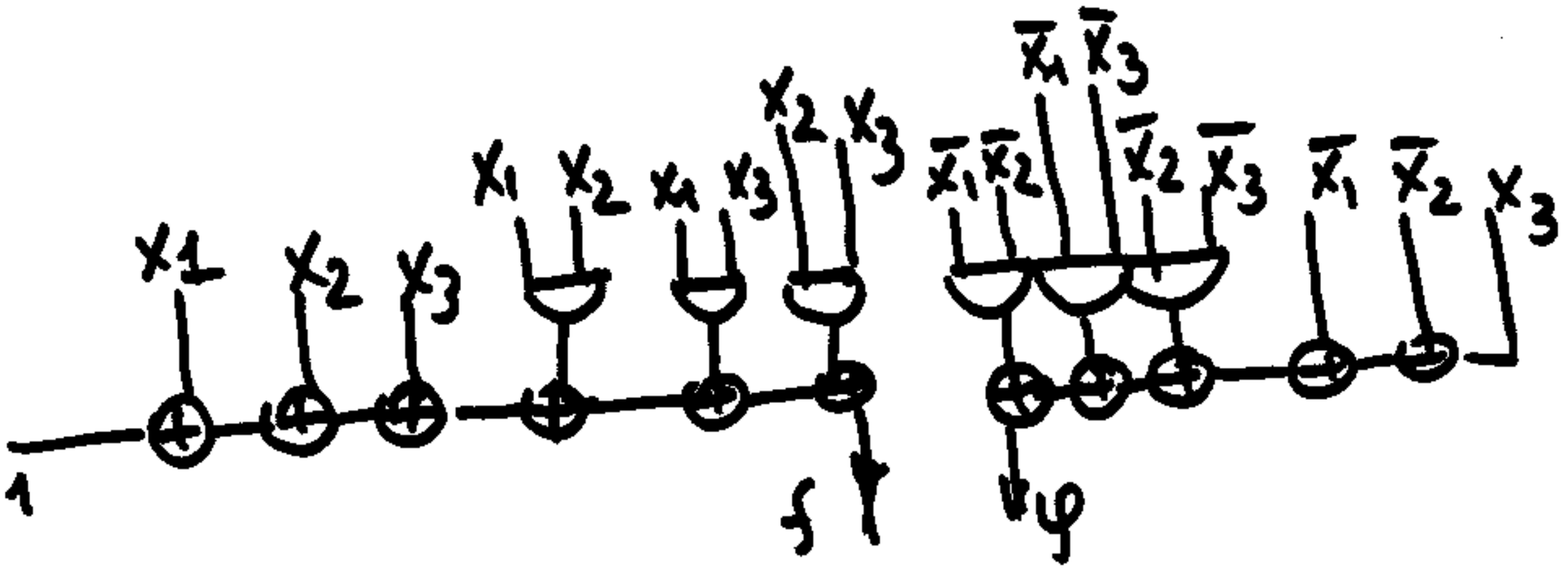
$C_{IN} = \{000, 111\}$

$f(x_1, x_2, x_3) = \overline{x}_1\overline{x}_2\overline{x}_3 + x_1x_2x_3 =$

$= (1 \oplus x_1)(1 \oplus x_2)(1 \oplus x_3) \oplus x_1x_2x_3 =$

$= 1 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_1x_2 \oplus x_1x_3 \oplus x_2x_3$

$g(x_1, x_2, x_3) = \overline{x}_1 \oplus \overline{x}_2 \oplus \overline{x}_3 \oplus \overline{x}_1\overline{x}_2 \oplus \overline{x}_1\overline{x}_3 \oplus \overline{x}_2\overline{x}_3$


x1 x2 x3
x1 x2 x4 x3
x2 x3
x1 x2 x3
x1 x2 x3
x1 x2 x3