Katinka Wolter
Inst. für Informatik
Freie Universität Berlin

Assignment 1. Faults of a full adder

Let us consider the full adder. We want to study the influence of functional and hardware faults on the behaviour of the circuit, and to determine the resulting failures.

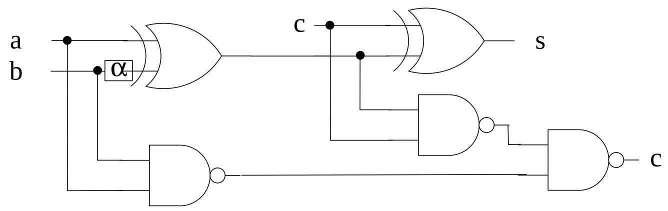

Figure 1: Adder

- Study the following functional fault, noted F_1, introduced during the design phase: the EXCLUSIVE OR gate has been transformed into an IDENTITY gate (its output is '1' if and only if the two inputs have the same value).
- Study the hardware fault, noted F_2, occurring during the operation: a 'stuck-at 0' noted α in figure 1.
- Compare the failures provoked by these two faults.

Assignment 2. Fault models and failures

Consider the circuit of Figure 2

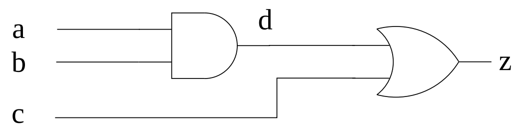

Figure 2: Redundant wire

1. Determine by formal analysis the failures provoked by each fault of the single stuck at 0/1 model. Draw and compare the resulting truth tables.
2. Make the same analysis with two other functional faults transforming the AND and OR gates.

Assignment 3. Single Parity Code

The code words of a parity code are obtained by adding a parity bit, i.e. such that the number of bits '1' in the code word is even. Consider the case where $m = 4$.

1. Find the code word of the useful word (1 0 1 1), and determine all detectable errors.
2. give an example of a non-detectable error.
3. Calculate the following characteristics of this code: *capacity, density, coverage rate* and *redundancy rate.*

Assignment 4. Hamming Code C(7, 4)

We consider a multiple parity detecting and correcting code such that: $k = 4$ and $n = 7$. The bits of the code word y are obtained from the word to be coded u in accordance with the following parity relations:

$$\begin{aligned} y_1 &= u_1 \oplus u_2 \oplus u_4, \\ y_2 &= u_1 \oplus u_3 \oplus u_4, \\ y_3 &= u_1 \\ y_4 &= u_2 \oplus u_3 \oplus u_4, \\ y_5 &= u_2, y_6 = u_3, y_7 = u_4 \end{aligned}$$

where u_i and y_i are the bits i of u and y.

1. Analyse this code and show that it detects all single errors **and** all double errors.
2. Show that this code **only** detects and corrects all single errors.
3. The definition of this code corresponds to a simple exchange of the relations given in class for the Hamming code C(7, 4). Compare the detecting and correcting capabilities of these two codes.
4. How can we modify this code such that it is able to detect all single and double errors **and** correct all single errors (without making any confusion between them)?

Assignment 5. Linear Code

Reconsider the previous exercise by regarding the *Hamming* code as a linear code.

1. Determine the matrices G and H.
2. Check the vector coding operation.
3. Analyse the error detection and correction with the help of the matrix product $H.W^T$.

Assignment 6. Berger Code

1. Draw the code word table of a Berger code with $m = 4$. Is this code optimal?
2. Show that this code allows the detection of every unidirectional error. We will analyse this error detection capability with an error that increases the number of '0' bits, first of all on X, then on R, then on the 2 parts. Then we will reason with an error that reduces the number of '0'.
3. Consider a code derived from a *Berger* code that requires the calculation of the number of '1' bits in X in order to formulate the redundant part R. This code strongly looks like the Berger code. Show that this code does not allow, however, the detection of unidirectional errors.