

Implementação de encoder/checker CRC-8



André Almeida 88960, Rui Santos 89293, T3G7

Solução para proteger a integridade de uma mensagem, implementado com um cyclic redundancy check, com o polinómio $b(x) = x^8 + x^5 + x^3 + x^2 + x + 1$.

A mensagem e o resultado do resto da divisão pelo polinómio em cima são enviados juntos pelo encoder, aquando do envio da mensagem, para quando esta for recebida, o checker realiza de novo o resto e compara-o com o recebido.

Se estes forem iguais, a mensagem não foi corrompida durante a “viagem” até ao recetor.

```
--level1
s0_4 <= a(0) XOR a(4);
s2_9 <= a(2) XOR a(9);
s6_7 <= a(6) XOR a(7);
s5_8 <= a(5) XOR a(8);
s1_3 <= a(1) XOR a(3);
s9_10 <= a(9) XOR a(10);
s10_11 <= a(10) XOR a(11);
s6_14 <= a(6) XOR a(14);
s4_7 <= a(4) XOR a(7);
s0_3 <= a(0) XOR a(3);
s12_15 <= a(12) XOR a(15);
s7_9 <= a(7) XOR a(9);
s5_13 <= a(5) XOR a(13);
s1_8 <= a(1) XOR a(8);
s5_10 <= a(5) XOR a(10);

--level2
s10_11_12_15 <= s10_11 XOR s12_15;
s0_4_2_9 <= s0_4 XOR s2_9;
s6_7_5_8 <= s6_7 XOR s5_8;
s1_8_13 <= s1_8 XOR a(13);
s0_3_5_8 <= s0_3 XOR s5_8;
s0_4_6_7 <= s0_4 XOR s6_7;
s1_3_5_13 <= s1_3 XOR s5_13;
s1_3_6_14 <= s1_3 XOR s6_14;
s1_3_2_9 <= s1_3 XOR s2_9;
s5_10_14 <= s5_10 XOR a(14);
s9_10_13 <= s9_10 XOR a(13);
s4_7_8 <= s4_7 XOR a(8);
s2_9_10_11 <= s2_9 XOR s10_11;

--level3
s7_9_10_11_12_15 <= s7_9 XOR s10_11_12_15;
s1_3_5_13_15 <= s1_3_5_13 XOR a(15);
s10_11_12_15_1_3_6_14 <= s10_11_12_15 XOR s1_3_6_14;
s6_7_5_8_12 <= s6_7_5_8 XOR a(12);
s1_3_6_7_5_8 <= s1_3 XOR s6_7_5_8;
s6_14_4_7_8 <= s6_14 XOR s4_7_8;
r(3) <= s0_4_2_9 XOR s1_8_13;
r(4) <= s1_3_2_9 XOR s5_10_14;

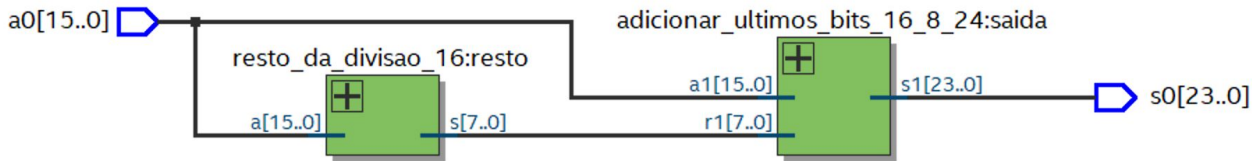
--level4
r(0) <= s0_3_5_8 XOR s7_9_10_11_12_15;
r(1) <= s0_4_6_7 XOR s1_3_5_13_15;
r(2) <= s0_4_2_9 XOR s10_11_12_15_1_3_6_14;
r(5) <= s0_4_2_9 XOR s6_7_5_8_12;
r(6) <= s1_3_6_7_5_8 XOR s9_10_13;
r(7) <= s2_9_10_11 XOR s6_14_4_7_8;
```

Encoder - Versão Paralela

O resto foi calculado utilizando as suas propriedades

XORs: 42

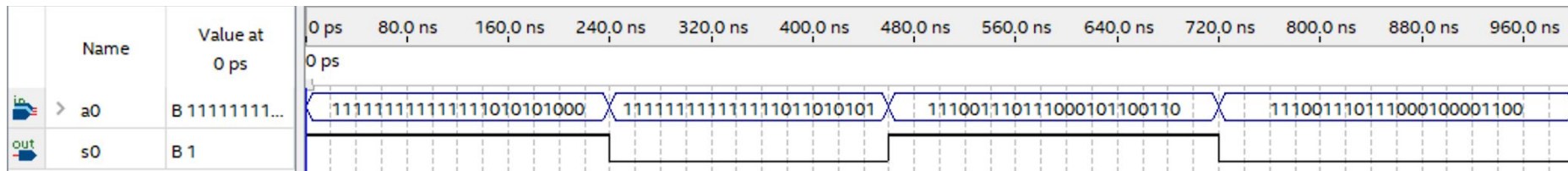
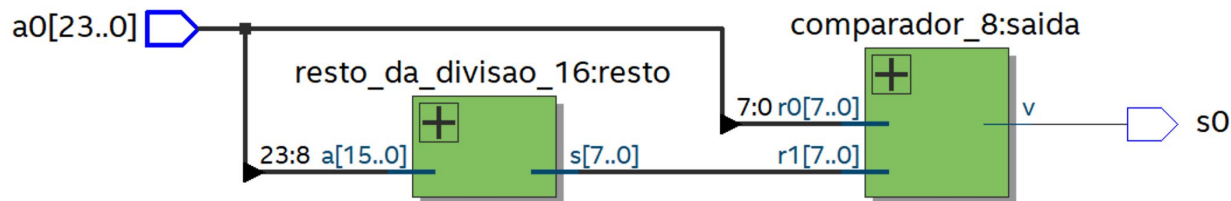
Atraso Máximo: 4

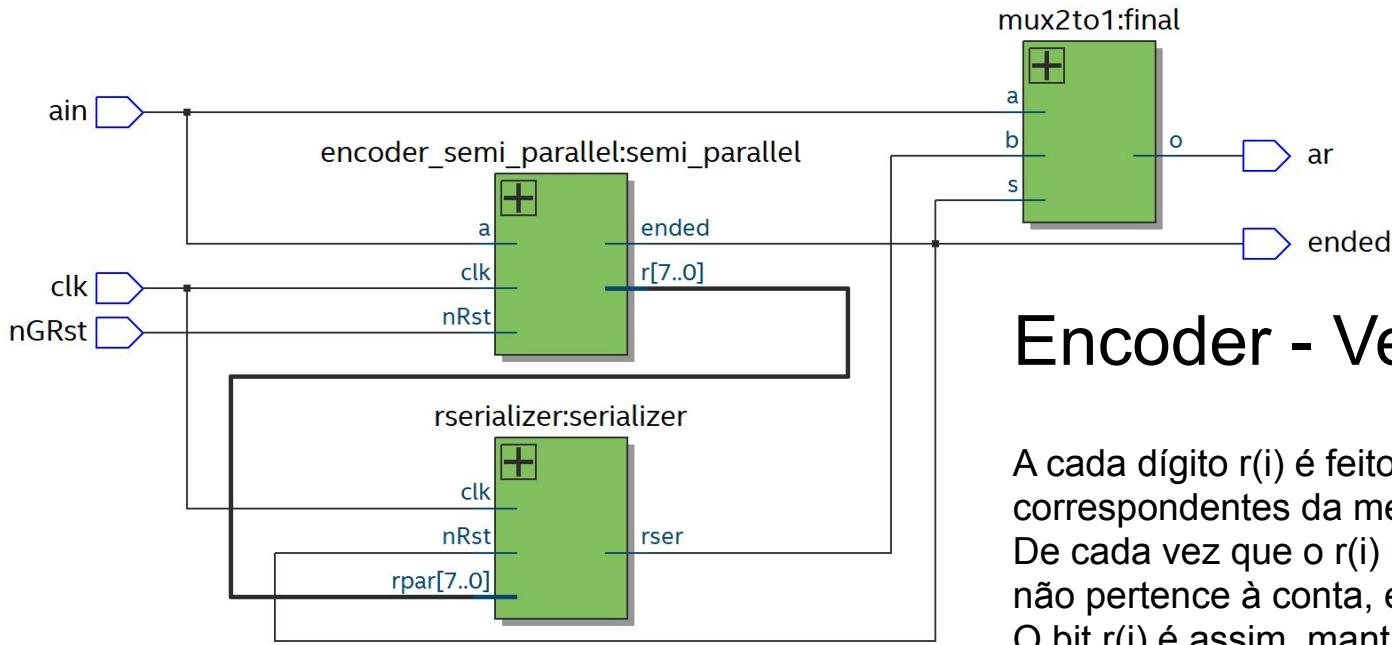


		Value at	0 ps	80,0 ns	160,0 ns	240,0 ns	320,0 ns	400,0 ns	480,0 ns	560,0 ns	640,0 ns	720,0 ns	800,0 ns	880,0 ns	960,0 ns
	Name	Value at	0 ps												
			0 ps												
	> a0	B 11111111...													
	> s0	B 11111111...													

Checker - Versão Paralela

É calculado o resto dos primeiros 16 bits de entrada (usando o mesmo módulo do encoder) e é comparado com os últimos 8 bits de entrada que correspondem ao resto enviado. O comparador envia 1 no caso das entradas serem diferentes e 0 no caso contrário. Para isso, usa 8 XORs e 7 ORs.





Encoder - Versão Série

A cada dígito $r(i)$ é feito um xor com os dígitos correspondentes da mensagem (a)
 De cada vez que o $r(i)$ recebe um bit do 'a' que não pertence à conta, este é posto a valor 0
 O bit $r(i)$ é assim, mantido, graças à propriedade $x \text{ xor } 0 = x$

