



# Guia de Verificação Bloco reed-solomon

Confidencial

---

Laboratório de Excelência em Microeletrônica

Virtus  
Universidade Federal de Campina Grande

---

---

Este é um documento **CONFIDENCIAL**  
é vetado qualquer reprodução ou divulgação não autorizada.

---

## Revision History

Revision	Date	Author(s)	Description
1.0	10.02.19	José Iuri	created

Confidencial

## Conteúdo

Confidencial

## 1 Introdução

### 1.1 Objetivos

Esse relatório tem o objetivo de apresentar o desenvolvimento de uma arquitetura de *testbench* para a verificação funcional baseada na metodologia UVM do bloco reed-solomon.

### 1.2 Temos Utilizados

Termo	Significado
DUT	Device Under Test
UVM	Universal Verification Methodology
RM	Reference Model
RTL	Register Transfer Model

### 1.3 Nomenclaturas e Representação de Bits e Amostras

Sample Representation	MSS	Most Significant Sample	(MSS:LSS)
	LSS	Least Significant Sample	
Bit Representation	MSB	Most Significant Bit	[MSB:LSB]
	LSB	Least Significant Bit	
	NBW	Total Number of Bits - Word length	
	NBI	Number of Integer portion length	
Fixed Point Representation	QUANT	Quantization by default: TRN	<WL,IWL[,QUANT]
	OVFLW	Overflow by default: WRAP	[,OVFLW][,NBITS] >bits
	NBITS	Number of saturated bits, only used for overflow mode and specifies how many bits will be saturated	
Logic Values	'1'	Bit asserted	
	'0'	Bit de-asserted	
	'x'	Bit don't care	
	R	Readable bit	
	W	Writable bit	
	U	Unimplemented bit, read as 0/1/x	
	-n	Value at POR	

### 1.4 Referências

## 2 DUT

### 2.1 Descrição

Xxxxxx xxxxxxxx.

### 2.2 Funcionalidades

- XXXXXXXX;
- yyyy;
- ZZZZZZZZZZ;

### 2.3 Parâmetros

Parâmetro	Tipo	Valor padrão	Descrição
NBW_OUT	integer	8	Número de bits de saída.

### 2.4 Interface

Sinal	Tamanho	I/O	Sync	Descrição
rst_async_n	1	I	Async	Reset, assíncrono e BAIXO ativo
clk	1	I	clk	Clock funcional

### 3 Arquitetura Proposta

Um conjunto de testes foi desenvolvido no intuito de verificar as funcionalidades do bloco de forma isolada. Os testes são aplicados por meio de um *testbench* escrito em *Systemverilog* baseado na metodologia UVM ilustrado na Figura ??.



# Diagrama

Figura 1: Arquitetura do testbench do reed-solomon

#### 3.1 Arquitetura do *testbench*

- Top

O nível superior do testbench cria instâncias do teste a ser executado, o DUT, o RM e a interface contendo os sinais que conduzirão valores de/para o DUT e o RM e os demais componentes do ambiente UVM criado.

- DUT

- RM
- Comparador
- Interface
- agent\_in
- agent\_out
- Test

### 3.2 Execução do *Testbench*

O ambiente de verificação pode ser obtido em:

```
commit 3213213213546510321358465 (origin/reed-solomon/Verification/tb)
Author: Fulano Silva <fulano.silva@embedded.ufcg.edu.br>
Date:   Tue Feb 12 11:11:04 2019 -0300
```

Último commit funcional.

Para rodar a simulação executar o Makefile. As opções de execução podem ser checadas no README.

```
options :
```

```
aqui fica o help do arquivo de README.
```

## 4 Testes

### 4.1 Simple Test

Teste simples com dados aleatórios de entrada de acordo com características previamente configuradas e entradas de configuração fixa.

## 5 Resultados

### 5.1 Simple Test

Resultados.