



Modelagem e Implementação em FPGA da Codificação 8b/10b

Victor Afonso dos Reis¹

Faculdade de Engenharia de Ilha Solteira, UNESP, Ilha Solteira, SP,

Departamento de Engenharia Elétrica, Unesp, Ilha Solteira, SP

Lucas Arruda Ramalho²

Faculdade de Engenharia de Ilha Solteira, UNESP, Ilha Solteira, SP,

Departamento de Engenharia Elétrica, Unesp, Ilha Solteira, SP

Ailton Akira Shinoda³

Faculdade de Engenharia de Ilha Solteira, UNESP, Ilha Solteira, SP,

Departamento de Engenharia Elétrica, Unesp, Ilha Solteira, SP

Resumo. Em comunicações de altas frequências, onde a confiabilidade é relevante, recomenda-se a codificação 8b/10b. Neste tipo de transmissão, pode ocorrer uma série de problemas. Um problema de extrema relevância em comunicações digitais é o sincronismo entre transmissor e receptor, que pode ser prejudicado por uma longa sequência de níveis lógicos, zeros (0's) ou um (1's), no canal de transmissão. Neste projeto modela-se e implementa-se um sistema da codificação 8b/10b, que soluciona o desbalanceamento de bits, no kit de desenvolvimento em FPGA da Xilinx(Kintex 7), através da linguagem de descrição de hardware VHDL pelo software VivadoTM. Pela implementação obteve-se um sistema genérico para o teste de codificações descritas em VHDL, além de parâmetros de frequência, taxa de dados e características do funcionamento da transmissão usando o sistema da codificação 8b/10b.

Palavras-chave. VHDL, FPGA, Codificação 8b/10b, Transmissões, Implementação.

1 Introdução

A complexidade dos sistemas de comunicação e as suas velocidades evoluíram ao longo do tempo, porém um modelo elementar é ilustrado na Figura 1. Em qualquer meio de transmissão, os dados transmitidos podem sofrer distorções ou interferências causadas por problemas no canal de transmissão [4].

¹victor.afonsoreis35@gmail.com

²lucasarrudaramalho@gmail.com

³shinoda@dee.feis.unesp.br

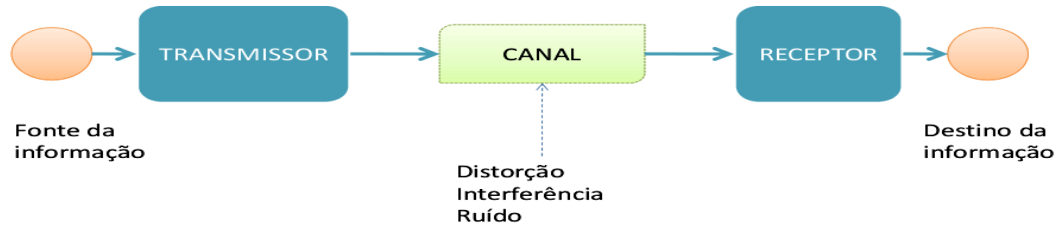


Figura 1: Modelo de um Sistema de Comunicação Elementar.

Em altas frequências os problemas comuns em qualquer meio de transmissão são mais evidentes, como por exemplo: a atenuação do sinal, dessincronização entre o transmissor e receptor além de ruídos apresentados no canal de transmissão. A dessincronização entre o transmissor/receptor e a aparição de dados ruidosos são um dos fatores que limita a velocidade dos sistemas [5].

A codificação 8b/10b é uma técnica de codificação em blocos, o qual adiciona-se redundâncias de forma lógica em um bloco de (m) bits gerando um bloco de (n) bits. Este mapeamento possui algumas características próprias da codificação, o qual possibilita a verificação de possíveis erros na transmissão. Algumas codificação em blocos realiza um balanço DC no dado, portanto permite facilitar a sincronização entre o emissor e receptor por meio de circuitos externos. Codificações como por exemplo a 4b/5b, a 6b/8b, a 8b/10b, a 64b/66b e a 128b/130b utilizam esse método [2].

O uso da codificação de blocos aumenta a qualidade do sistema de transmissão, apesar de diminuir a taxa de transmissão de dados efetiva [3]. Dessa maneira, o desafio resume-se em torno da análise dos parâmetros de cada codificação, para ser capaz de fornecer uma solução de menor custo e que atenda os requisitos do sistema [1]. Sendo assim, o objetivo deste trabalho é modelar e implementar em *VHSIC Hardware Description Language* (VHDL) um sistema com a codificação 8b/10b, dentro de um kit de desenvolvimento em FPGA da Xilinx (Kintex 7 KC705). Através dessa implementação, é possível obter informações e parâmetros do sistema: frequência de operação máxima, taxa de transferência de bits por segundo e o comportamento da codificação ao inserir erros na transmissão.

Pela análise parcial da simulação do sistema no FPGA, foi possível obter-se características de sincronismo próprios da codificação ao trabalhar com erros além de informações sobre a velocidade do sistema. A seção 2 resalta a teoria da codificação 8b/10b. A seção 3 apresenta a modelagem da codificação 8b/10b em máquina de estados além de uma ilustração do sistema implementado no FPGA. A seção 4 descreve a implementação no FPGA do sistema descrito em VHDL. A seção 5 conclui os resultados da implementação no FPGA do sistema da codificação 8b/10b.

2 Codificação 8b/10b

Os dados de 8 bits são mapeados em 10 bits, dessa maneira alguns dados de 10 bits com o mesmo número de bits 1's e 0's são usados como dados de controle da transmissão [?]. Na codificação é descrito o conceito de disparidade, apresentando dois tipos: a disparidade do

sistema, ou do inglês *Running Disparity* (RD), e a disparidade dos dados. A disparidade é definida como a diferença entre o números de bits 1's e 0's do dado codificado para 10 bits.

A codificação é descrita de forma que os dados transformados em 10 bits só possuam disparidade de dados +2, -2 ou 0 (nula). Desta maneira, a disparidade dos dados é obtida no momento em que se codifica o dado de entrada para 10 bits. A disparidade do sistema, ou RD, é obtida seguindo regras de acordo com a disparidade dos dados e também é responsável pela escolha dos dados no momento da codificação [1].

O RD alterna-se entre +1 e -1 ou permanece o mesmo, de acordo com as regras, das disparidades dos dados, ilustradas na Figura 2. Quando o RD for -1 não é possível a disparidade dos dados ser -2, assim como se o RD for +1 é impossível a disparidade dos dados ser +2. Nestes casos, ao se obter esse tipo de disparidade para cada RD mencionado no lado do receptor representa um erro na transmissão.

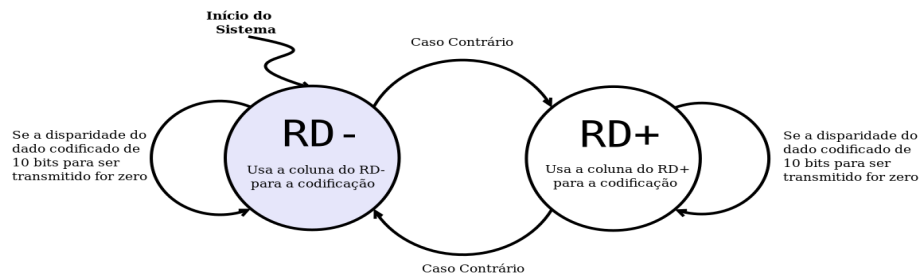


Figura 2: Máquina de estados para a transição do RD.

A característica da disparidade de dados codificados possuir sempre valores definidos (+2, -2 ou 0), possibilita a detecção de erro pelo receptor. Como descrito, para codificar os dados usa-se o RD que sempre será -1 quando o sistema for inicializado. [?]. Na figura 3 é ilustrado o modo como a codificação dos dados é realizada. O dado 8 bits é separado em duas partes, codificando-os e realizado a junção gerando um dado de 10 bits como ilustrado.

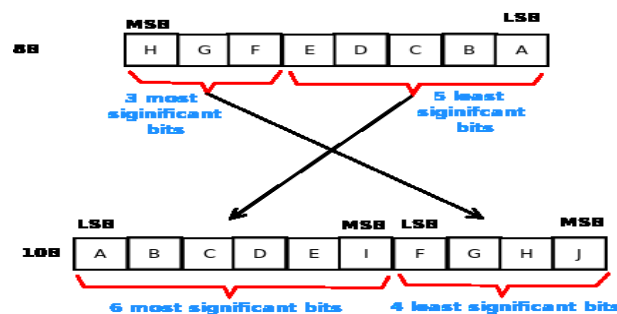


Figura 3: Esquema da separação para codificação 8b/10b..

Pela descrição, a codificação promove um balanceamento DC no sinal, ou seja, o dado a ser transmitido não possui níveis lógicos altos ou baixos por muito tempo. Esse balanço

DC torna-se importante para a recuperação do relógio e consequentemente sincronização entre emissor e receptor.

3 Descrição e Modelagem em Máquinas de Estado em VHDL do Sistema da Codificação 8b/10b

A modelagem da codificação 8b/10b pode ser feita por meio de máquinas de estados no VHDL, representando o comportamento de um circuito ou programa por diagramas.

O funcionamento da codificação 8b/10b, descrita em VHDL através de máquinas de estados, é ilustrado na figura 4. Como pode-se analisar há a presença de sete estados no sistema do *encoder* e no *decoder*.

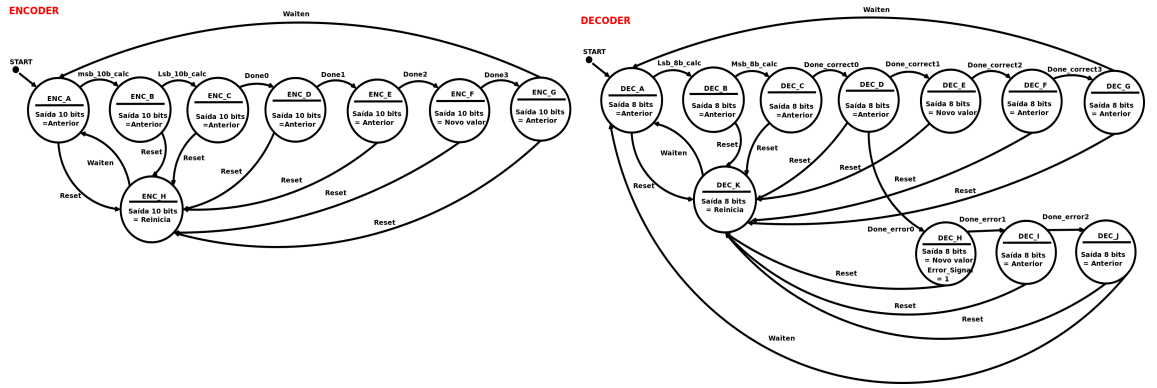


Figura 4: Máquina de estados do sistema implementado a codificação 8b/10b em VHDL.

No *encoder* o estado "ENC_A" refere-se ao estado de espera do dado de 8 bits na entrada. Já o estado "ENC_B" e "ENC_C" trata a codificação da parte menos e mais significativa do dado de entrada (8 bits), como na figura 3. Nos estados "ENC_D", "ENC_E", "ENC_F" e "ENC_G" realiza-se a junção das partes codificadas formando a saída de 10 bits, cálculo da disparidade de dados, atualização da saída e atualização do RD, respectivamente em cada estágio.

A máquina de estado do *decoder* apresenta comportamento similar, apesar de inverso, com o objetivo de decodificar os blocos do dado codificado. Assim a mesma é capaz de fornecer o dado de 8 bits decodificado e caso no estado "DEC_D" identifique que houve erro, vai para os estados "DEC_H", "DEC_I" e "DEC_J". Nestes estados a saída vai para alta impedância e um sinal de erro vai para nível lógico alto. Após o estado "DEC_J" o sistema volta para o estado de espera "DEC_A" para receber um novo dado de 10 bits.

A modelagem da máquina de estados foi descrita no software VivadoTM na linguagem VHDL, assim como o sistema para sua implementação no FPGA. Desta forma, além da descrição do *encoder* e *decoder*, há a necessidade de descrever todos os sistemas periféricos necessários para o funcionamento da codificação. O sistema da codificação e os demais periféricos são interligados, descritos e referenciados dentro de um arquivo denominado *Top File*. A descrição do sistema para implementação no FPGA é ilustrado na figura 5.

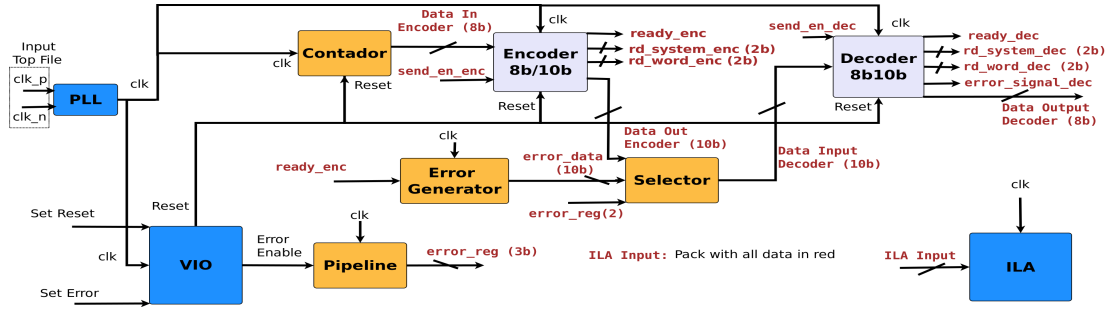


Figura 5: Descrição do Top File da codificação 8b/10b para implementação no FPGA.

Pela figura 5 observa-se para o funcionamento do sistema deve-se descrever um *Phase-locked Loop* (PLL), um *Virtual Input Output* (VIO), um *Integrated Logical Analiser* (ILA), um contador para inserir dado no *encoder*, um sistema (VIO, Pipeline) para inserir o erro no sistema e um arranjo para modificar o dado de saída do *encoder* para ser entrada no *decoder*.

Através do PLL insere-se o clock o qual FPGA é capaz de implementar. O VIO permite inserir sinais dentro do sistema, como por exemplo o *reset* e o sinal de erro. O sistema de pipeline seta um sinal de erro em somente um ciclo de *clock*. O sistema *Error Generator* gera um vetor de 10 bits, com apenas um bit em nível lógico alto, rotacionando-os quando o sinal *ready_enc* for alto. O bloco seletor realiza uma operação XOR com o dado de 10 bits do *Error Generator* e o dado de saída do *encoder*, somente quando o sinal "error_reg(2)" for alto. Caso contrário, o bloco seletor deixa passar somente o dado de saída do *encoder*, ou seja, sem passar pela operação XOR. Pelo ILA pode-se visualizar todos os sinais dentro do sistema, dessa forma todos os dados marcados em vermelho na figura 5 são inseridos no ILA para obter suas formas de onda.

4 Implementação e Simulação no Kit FPGA da Xilinx (Kintex 7)

Após a implementação através do software Vivado™, a frequência de *clock* definida no teste do sistema foi de 300 MHz. Esta frequência não é de operação máxima. Dessa forma, o sistema nessa faixa de frequência opera a uma taxa de transmissão em torno de 2,4 Gb/s. O ILA e VIO impedem a determinação da frequência máxima, uma vez que estes sistemas não suportaram a elevação da taxa de *clock*. Dessa forma, testou-se retirar o ILA e a máxima frequência obtida foi de 400MHz, ou seja, uma taxa de dados transmitidos de 3,2 Gb/s. A única opção para remoção de algum componente do sistema era o ILA, pois o sistema depende do VIO para o funcionamento.

Após o teste da implementação obteve-se uma latência de codificação, em ciclos de *clock*, de 5 ciclos até o dado de entrada estar devidamente disponível na saída. Para a decodificação, esta latência é de 4 ciclos totalizando 9 ciclos para codificação e decodificação. Porém, o sistema que insere erro na transmissão utiliza 2 ciclos de *clock* totalizando 11

ciclos para o dado codificado estar devidamente decodificado na saída.

Na figura 6 é ilustrado a simulação do sistema no kit FPGA da Xilinx. O sinal *Error Input*, gerado pelo usuário, permanece em nível lógico alto até que seja inserido dado no *decoder* gerando erro em um bit no dado. Após gerado o erro, 4 ciclos de *clock* depois o sistema fornece um sinal de erro. O RD entre *encoder* e o *decoder* entra em descompasso e recupera-se após 33 ciclos de *clock*, ou seja, após 3 palavras na transmissão. Este descompasso do RD entre o *encoder* e *decoder* é característico da codificação e faz com que algumas palavras transmitidas sejam perdidas. Após alguns testes obteve-se um descompasso máximo de 88 ciclos, ou seja, 8 palavras perdidas na transmissão.

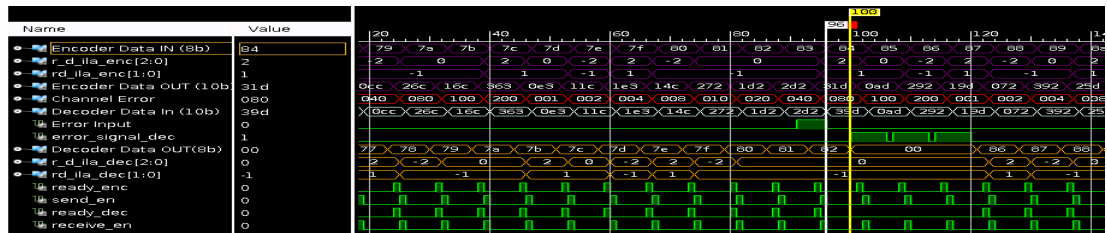


Figura 6: Simulação da codificação 8b/10b implementada no FPGA com erros inseridos no canal.

Após alguns testes inseriu-se um erro no dado, porém dado era decodificado para um valor errôneo. Após algumas palavras era gerado um sinal de erro pelo sistema. Isto acontece pelo fato de embora o *encoder* e o *decoder* desbalancearem os RD's, alguns dados possuem a mesma palavra para qualquer um dos RD's. Dessa forma, pelos testes a latência máxima para encontrar o erro na transmissão foi de 53 ciclos. Na figura 7 é ilustrado este caso. Observa-se que o erro não foi detectado e só posteriormente, pelo desbalanço do RD entre os dois sistemas, foi gerado um sinal de erro.

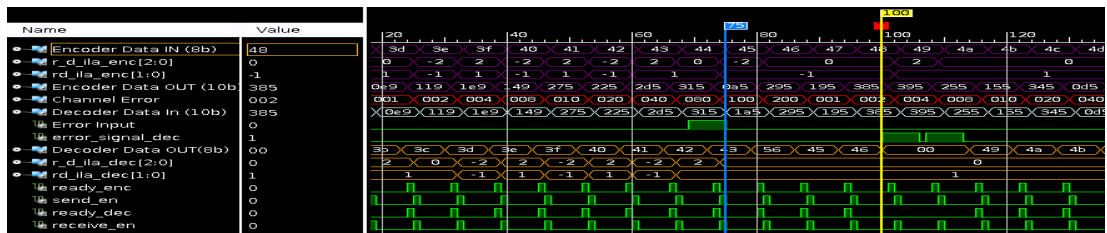


Figura 7: Simulação do sistema da codificação 8b/10b com erro não identificado.

Pela figura 7, observa-se que o dado em hexadecimal "46" foi inserido e posteriormente saiu o dado também em hexa "56". No ciclo 75 é inserido o dado em hexadecimal "45" e posteriormente sai o mesmo dado, dessa forma mesmo com o sistema com o RD em desbalanço o dado foi decodificado certo. Só alguns ciclos de *clock* posteriores foi gerado um erro por conta do desbalanço dos RD's.

5 Conclusão

A utilização de sistema codificadores para comunicação de alta velocidade é vital para sincronismos entre transmissão e recepção, bem como verificação de erros no canal. A escolha entre as codificações é realizada analisando o custo benefício entre robustez, confiabilidade, ocupação de taxa de dados e largura de banda, e qualidade do canal utilizado.

Este artigo efetuou uma implementação da codificação 8b/10b em um kit de desenvolvimento em FPGA da Xilinx (Kintex 7). Os resultados mostram que a codificação pode apresentar falha ao detectar erros, por conta do desbalanço do RD entre *encoder* e *decoder*. Entretanto, o sistema da codificação 8b/10b implementada no kit possui uma latência máxima de codificação e decodificação de 11 ciclos de *clock*. A máxima frequência de *clock* obtida pelo sistema foi de 400 MHz, com uma taxa de transferência de 3,2 Gb/s.

O sistema descrito e implementado no FPGA para a codificação 8b/10b, pode ser utilizado como base para a implementação e simulação no FPGA de outras codificações. Logo este trabalho também descreve máquinas de estado para o sistema da codificação 8b/10b em VHDL. Pretende-se em trabalhos futuros, descrever um sistema que realize a análise estatística dos dados, obtendo a robustez do sistema implementado no FPGA.

Agradecimentos

Os autores agradecem à FUNDUNESP e ao SPRACE pelo apoio.

Referências

- [1] A. X. Widmer and P. A. Franaszek, "A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code," in IBM Journal of Research and Development, vol. 27, no. 5, pp. 440-451, Sept. 1983. DOI: 10.1147/rd.275.0440
- [2] B. A. FOROUZAN, "Digital Transmission", In: B. A. FOUROUZAN, "Data Communications and Networking", 4. ed, New York: Mcgraw-hill Education, (2007). Cap. 10. p. 267-306.
- [3] E. Berlekamp, R. Peile and S. Pope, "The application of error control to communications", in IEEE Communications Magazine, vol. 25, no. 4, pp. 44-57, April (1987). DOI: 10.1109/MCOM.1987.1093590.
- [4] F. Borges, Transmissão de Dados, Carnaxide: Schneider Electric, 2008. Disponível em: http://www.schneiderelectric.pt/documents/product-services/training/transmissao_dados.pdf. Acesso em: 1 ago. 2017.
- [5] T. D. PRA, Medidor de Taxa de Erro de Bit para Fibra Óptica. 2012. 76 f. TCC (Graduação) - Curso de Engenharia Elétrica, Engenharia Elétrica, Universidade Federal do Rio Grande do Sul, Porto Alegre, (2012). Disponível em: <https://www.lume.ufrgs.br/bitstream/handle/10183/65425/000858235.pdf?sequence=1>. Acesso em: 1 ago. 2017.