

Projeto 1- PSI

Use o material de apoio do arquivo PDF, Implementando_LFSRs _Galois_18.

Data de Entrega de Relatório: **11 de maio de 2018**

Módulo Random_num

O Módulo VHDL *Random_num* a ser projetado é um gerador de números aleatórios que deverá ser integrado ao Circuito do Snake, substituindo a função de geração de números aleatórios testada junto ao bloco *Num Gen* (aula 4).

A figura 1 ilustra as interfaces do módulo *Random_num* e a sua esperada conexão dentro do datapath do Snake. A figura 1.a mostra o esquema utilizado na Aula 4, com a função *rand_num_f* sendo chamada dentro do bloco *Num Gen*; já a figura 1.b indica como deve ser a nova configuração, uma vez que haverá um módulo específico para a geração do número.

Dos $n(=12)$ bits do LFSR gerador do número aleatório a ser projetado, apenas 6 serão utilizados para a definição do endereço de memória do Snake, na forma indicado pela figura. Após a concatenação com dois bits de valor '0', um sinal de 8 bits é enviado ao módulo *Num Gen* (obs. é assim no Snake apesar de que nas simulações da aula 4, o número de bits usado foi 6).

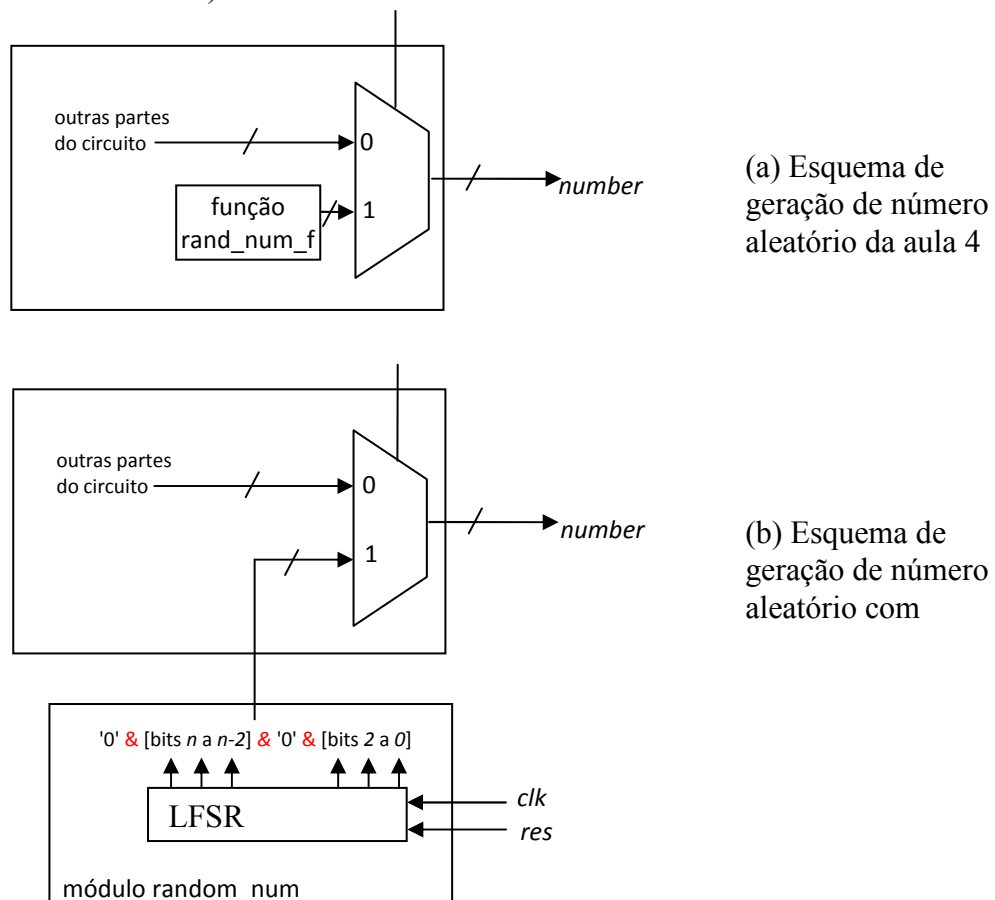


Figura 1

O reset (res) do circuito deve ser utilizado para colocar a saída dos FFs em "1" (impondo o seed), na forma da Figura 2.

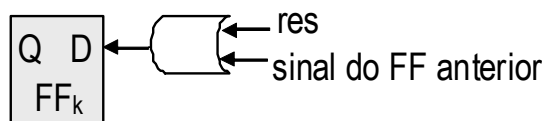


Figura 2

O trabalho consiste de 5 etapas.

1. Calcular os coeficientes do polinômio gerador do seu LFSR.
2. Simular o LFSR de forma a compreender e conhecer a sua sequência pseudoaleatória.
3. Escrever o código VHDL estrutural de um *linear feedback shift register* (LFSR) genérico e personalizar o código para o seu polinômio utilizando **obrigatoriamente** o comando GENERATE.
4. Demonstrar que a sequência obtida é a correta por simulação.
5. Preparar um relatório descritivo das etapas e conclusões.

Importante: Cada aluno deverá personalizar o LFSR de acordo com o seu #USP.

Abaixo estão as características do projeto.

- a) O LFSR será do tipo X como descrito no material adicional, correspondendo a um polinômio de grau 12, da forma $(x^{12} + a_{11} \cdot x^{11} + \dots + a_1 \cdot x^1 + 1)$. Os coeficientes $a_{11} \dots a_1$, serão definidos de acordo com o #USP de cada aluno.
- b) Cálculo dos coeficientes do polinômio:
 - Calcule $X = \#USP \bmod 2048$ (resto da divisão $\#USP/2048$).
 - Sendo X um número decimal converta-o em número binário de 11 *bits*. O dígito binário mais significativo corresponderá ao coeficiente c_{11} e o menos significativo ao coeficiente c_1 . Vamos dar dois exemplos.
 - Exemplo 1:
 - $X_{\text{decimal}} = 53 \rightarrow X_{\text{binario}} = 110101$
 - Polinômio $\rightarrow x^{12} + x^6 + x^5 + x^3 + x^1 + 1$
 - Exemplo 2:
 - $X_{\text{decimal}} = 2045 \rightarrow X_{\text{binario}} = 1111111101$
 - Polinômio $\rightarrow x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^1 + 1$

Atenção: Caso $X=0$, adote os 3 dígitos menos significativos do seu #USP como seu valor de X.

Relatório do projeto: o aluno deve apresentar o polinômio, explicando como chegou à configuração (atenção: revise e confirme que o polinômio obtido é o correto; em caso de erro, haverá comprometimento dos demais resultados e, portanto da nota).

- c) Conhecido o seu polinômio, deve-se estruturar o circuito LFSR composto de DFFs e XORs.

Relatório do projeto: apresentar um esboço (em forma digital ou manuscrito) do esquema do circuito LFSR desenvolvido.

- d) Gere a sequência de bits gerados através do software on-line descrito no documento de descrição do LFSR Galois. Capture as imagens do software com os resultados dos 10 primeiros ciclos após a inicialização (em '1111...111') para serem apresentados.

Relatório do projeto: impressão das imagens de tela com os resultados da simulação (10 ciclos).

- e) Faça a descrição equivalente do circuito em VHDL (figura 1.b) seguindo o esquema dado no item c) (atenção: utilize os mesmos nomes para os sinais e portas).

- O DFF contendo um sinal de *set* é fornecido ao aluno (na área da disciplina no NEWSERVERLAB).

- Para o XOR, use o módulo utilizado para os somadores das aulas anteriores.

- Você deverá usar obrigatoriamente o comando GENERATE para construir o conjunto do LFSR. O comando deverá ser usado de forma "inteligente" de forma a otimizar a codificação.

- O vetor de estados conterá os bits de saída dos FFs (dos bits 12 a 0) e portas do módulo deverão ser de acordo com o especificado na figura 1.b.

Relatório do projeto: incluir a descrição do projeto em VHDL.

- f) Codifique um testbench para validar o seu código VHDL através do simulador ModelSim. Prepare uma carta de tempos no **Wave** de forma a mostrar uma sequência de estados (com o vetor de estados). Simule pelo menos por vinte (20) ciclos de relógio de evolução da sequência, após a ativação do sinal set.

Relatório do projeto:

1) incluir a descrição VHDL do *testbench* com seus componentes.

2) impressão legível¹ da carta de tempos com pelo menos 20 ciclos de relógio (deixe os sinais importantes evidentes). Apresente tanto os sinais dos estados(12 bits) como de saída (8 bits).

Obs. Represente os números aleatórios gerados com a mesma base numérica utilizada em **d**).

- g) Validação por comparação entre o resultado de seu projeto e o do software.

Relatório do projeto: Discuta e demonstre que (se) os resultados da simulação do projeto e da execução do software rodado são os mesmos (compatíveis entre si).

¹ Cartas de tempo não legíveis ou sem indicações claras serão consideradas como não entregues.