

Universidade Federal de Santa Catarina

EEL7123/EEL510269

Semestre: 2019/2 – Lab2a

Somadores RNS modulus  $\{2^{2n}, 2^n - 1, 2^n + 1\}$

## 1 Introdução e objetivos

O objectivo deste laboratório consiste em projetar em FPGA uma unidade aritmética somadora RNS vistas nas aulas teóricas. Estas unidades serão reutilizadas nas seguintes aulas experimentais para o desenvolvimento de unidades RNS completas com funcionalidade aritmética soma e multiplicação. A Figura 1 descreve os três níveis de operação das unidades RNS usando o conjunto de módulos  $\{m_1, m_2, m_3\} = \{2^{2n}, 2^n - 1, 2^n + 1\}$ : i) Conversores binário a RNS (Binary-to-RNS converters) vistas na aula 1a, ii) unidades aritméticas RNS (RNS arithmetic units) sobre a qual trabalharemos em esta aula e nas próximas 2 aulas de laboratório 2b e 3a e iii) conversores RNS a binário (RNS-to-Binary converters) vistas na aula 1b.

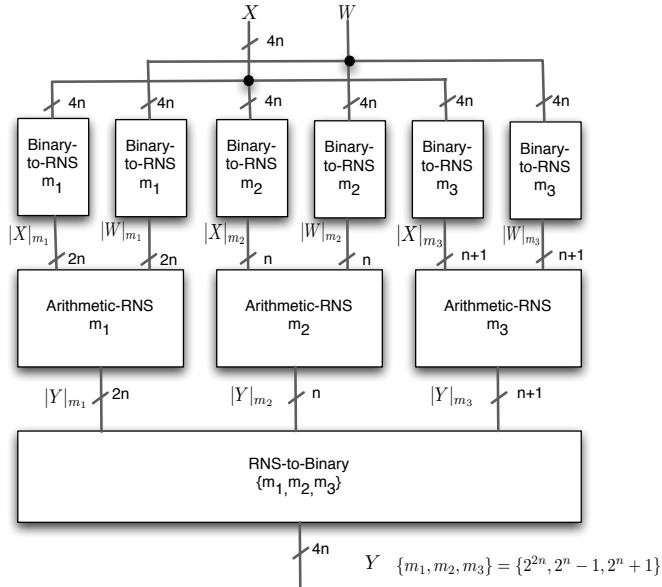


Figura 1: Unidade RNS completa usando conjunto de módulos  $\{m_1, m_2, m_3\} = \{2^{2n}, 2^n - 1, 2^n + 1\}$ .

## 2 Somador RNS

As operações soma em RNS são feitas por canal residual. De esta forma, para o moduli set usado nas aulas anteriores  $\{m_1 m_2 m_3\} = \{2^{2n}, 2^n - 1, 2^n + 1\}$ . Três somadores são necessários de modo a obter a representação do RNS, um para cada elemento de base.

- **Canal  $m_1 = 2^{2n}$ :** O canal mais simples é o conversor usando o modulo  $m_1$ . O valor da soma  $S_1 = |Y|_{m_1} = |R_1 + Q_1|_{m_1}$ , onde  $R_1 = |X|_{m_1} = \{r_{1,(2n-1)}, \dots, r_{1,0}\}$  e  $Q_1 = |W|_{m_1} = \{q_{1,(2n-1)}, \dots, q_{1,0}\}$  pode ser obtido por médio de truncar o valor de acarreo da soma, uma vez que  $|2^{2n}|_{2^{2n}} = 0$ .
- **Canal  $m_1 = 2^n - 1$ :** Em este caso o valor da soma  $S_2 = |Y|_{m_2} = |R_2 + Q_2|_{m_2}$ , onde  $R_2 = |X|_{m_2} = \{r_{2,(n-1)}, \dots, r_{2,0}\}$  e  $Q_2 = |W|_{m_2} = \{q_{2,(n-1)}, \dots, q_{2,0}\}$  pode ser obtido por médio de fazer uma rotação do acarreo da soma, uma vez que  $|2^n|_{2^n-1} = 1$ . No entanto usaremos a abordagem apresentada no slide 23 do capítulo 6\_7 das aulas teóricas.
- **Canal  $m_1 = 2^n + 1$ :** Em este caso o valor da soma  $S_3 = |R_3 + Q_3|_{m_3}$ , onde  $R_3 = |X|_{m_3} = \{r_{3,(n)}, \dots, r_{3,0}\}$  e  $Q_3 = |W|_{m_3} = \{q_{3,(n)}, \dots, q_{3,0}\}$  pode ser obtido por médio de fazer uma rotação do acarreo da soma com complemento, uma vez que  $|2^n|_{2^n+1} = -1$  (incluindo um factor corrector da mesma forma que fizemos no laboratório 1a para o canal  $2^n + 1$ ). No entanto usaremos a abordagem apresentada no slide 23 do capítulo 6\_7 das aulas teóricas.

## 3 Implementação em VHDL do somador RNS

- Com a placa em funcionamento, baixe o arquivo "*ProjetoLab02a.zip*" disponível no site da disciplina e descompacte esse arquivo na pasta "*EEL510269/lab02a*". Atenção: o caminho do diretório para o qual o arquivo será descompactado não deve conter espaços.
- Agora, execute o software Quartus II 13.0sp1 Web Edition (a versão 12.0sp2 também pode ser utilizada). Com o software em funcionamento, acesse o menu File e a opção Open File para abrir os três arquivos VHDL *Adder\_2\_2n.vhd*, *Adder\_2\_n\_neg1.vhd* e *Adder\_2\_n\_pos1.vhd* disponíveis na pasta destino da descompactação. Os VHDL correspondem com os somadores RNS modulo  $\{2^{2n}, 2^n - 1, 2^n + 1\}$ .

### 3.1 Tarefa a ser realizada na sala de aula

Agora o aluno deve preencher partes do código VHDL dos três canais modulares para assim obter um somador RNS usando o conjunto de modulos  $\{2^{2n}, 2^n - 1, 2^n + 1\}$  e  $n = 4$ .

**Para a implementação do canal  $m_1 = \{2^{2n}\}$**  abra o arquivo *Adder\_2\_2n.vhd* e descreva a implementação entre as linha de código 33 a 36.

**A implementação em VHDL dos somadores modulo  $m_2$  e  $m_3$  já são fornecidos completamente** *Adder\_2\_n\_neg1.vhd* e *Adder\_2\_n\_pos1.vhd*. A tarefa a seguir consiste na implementação do sistema completo Binário-RNS, soma

RNS e volta a binário como indicado na Fig. 1. Para isso vamos precisar de incluir um registrador para introduzir a componente  $X$  e  $W$  de forma sequencial. Um registrador de 4 bits é fornecido no site da disciplina *D\_4FF.vhd* com reset assíncrono (RST) e enable (EN). **O aluno deverá estender a 16 bits este registrador e incluir o sinal de Enable.** Uma vez feita dita extensão, o seguinte passo consiste em descarregar o projeto da aula 1b e incluir os somadores, os registradores e os três conversores binário-RNS para a entrada  $W$ . A Fig. 2 apresenta o mapa de pinos de entrada e saída para dita implementação.

Use o sinal *CLOCK\_50* como relógio do registrador. Visando evitar problemas de temporização em função do apertado de um KEY por um ser humano durar muitos ciclos de *clock*, o *Button Press Synchronizer* (ButtonSync) será fornecido em conjunto com o projeto deve ser utilizado. O *ButtonSync* converte apertos das KEYS em pulsos com período de um ciclo de clock. Assim, em seu projeto, as KEYS devem ser ligadas nas estradas do *ButtonSync*, e as saídas BTN(0) a BTN(3) do *ButtonSync* deverão ser utilizadas para controlar o registrador, o botão de pressão BTN(0) como reset assíncrono, e BTN(1) como botão de enable para introduzir a entrada  $W$ .

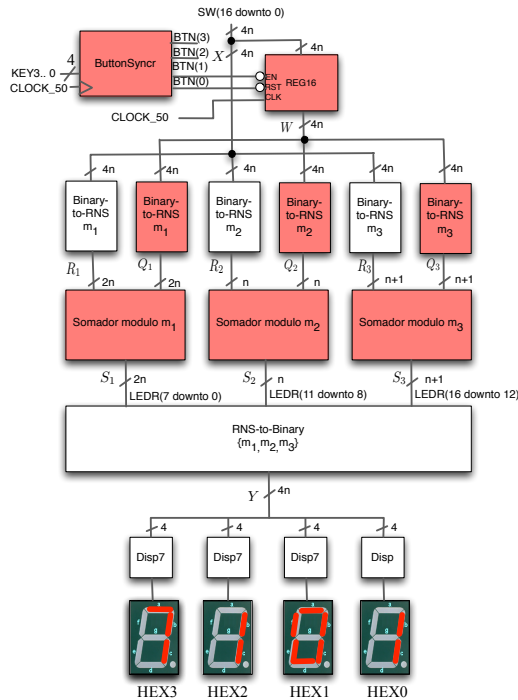


Figura 2: Bloco RNS com soma aritmética com associação de pinos entrada-saída.

Uma vez compilado sem erros abra modelsim e simule o circuito. Dica: use o script .do para forçar as entradas. **Preencha a tabela 1 com os dados da simulação.**

Tabela 1: Tabela de resultados de simulação

$R_1$	$R_2$	$R_3$	$Q_1$	$Q_2$	$Q_3$	$S_1$	$S_1$	$S_1$	$Y$
0	0	0	0	0	0				
255	1	1	255	1	1				
252	0	0	255	1	1				
162	10	5	252	0	0				

Tabela 2: Tabela de resultados na placa DE2

$R_1$	$R_2$	$R_3$	$Q_1$	$Q_2$	$Q_3$	$S_1$	$S_1$	$S_1$	$Y$
0	0	0	0	0	0				
255	1	1	255	1	1				
252	0	0	255	1	1				
162	10	5	252	0	0				

Uma vez terminada a simulação implemente na placa DE2 o circuito e preencha a tabela 2 com os resultados obtidos nos LEDs vermelhos. Importante: antes de desligar o computador, guarde a pasta lab2a em um pendrive ou envie por e-mail já que este circuito será usado nos seguintes laboratórios.

### 3.2 Questões finais

- **Pergunta 1:** Qual é o benefício de fazer a soma em RNS em comparação com binário?.

- **Pergunta 2:** O que esperas obter quando somas  $X = W = 2^{16} - 2^8 - 1$ ?