

Universidade Federal de Santa Catarina

EEL7123/EEL510269

Semestre: 2019/2 – Lab1b

## Conversor RNS-binário

### 1 Introdução e objetivos

O objectivo deste laboratório consiste em projetar em FPGA uma unidade conversora de numeração residual (RNS) a binário 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) já vistas no laboratório 1a, ii) unidades aritméticas RNS (RNS arithmetic units) que serão vistas nas aulas 2a, 2b e 3a e iii) conversores RNS a binário (RNS-to-Binary converters) que será visto nessa aula 1b.

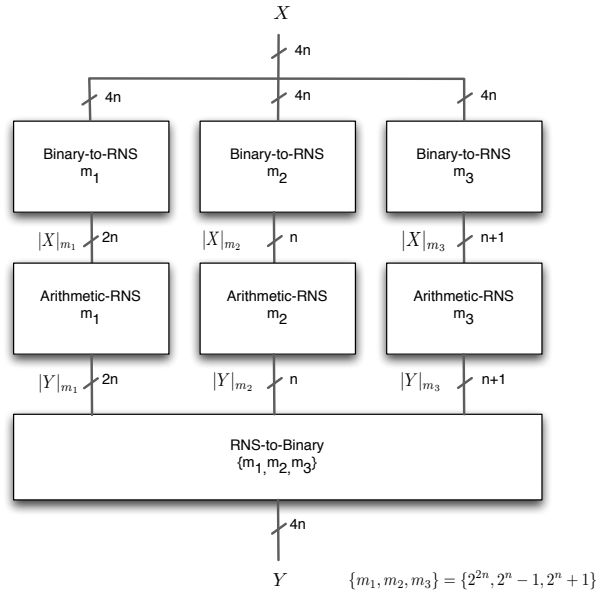


Figura 1: Unidade RNS completa usando conjunto de módulos  $\{m_1, m_2, m_3\} = \{2^n, 2^n - 1, 2^n + 1\}$ .

## 2 Conversor RNS-binario

A conversão de RNS a binario é obtida usando o Chinese Remainder Theorem (CRT), visto na aula teorica.

$$X = \left| \sum_{i=1}^3 \hat{m}_i \left| \hat{m}_i^{-1} \right|_{m_i} R_i \right|_M, \quad (1)$$

onde  $\hat{m}_i = \frac{M}{m_i}$ ,  $\left| \hat{m}_i^{-1} \right|_{m_i}$  é a multiplicativa inversa de  $\hat{m}_i$  e  $R_i$  as entradas residuais. As multiplicativas inversas são números inteiros que satisfazem a condição  $\left| \left| \hat{m}_i^{-1} \right|_{m_i} \hat{m}_i \right|_{m_i} = 1$ . Para o moduli set da aula anterior  $\{m_1, m_2, m_3\}$ , onde  $m_1 = 2^{2n}$ ,  $m_2 = 2^n - 1$ ,  $m_3 = 2^n + 1$  escolhido obtemos  $\hat{m}_1 = 2^{2n} - 1$ ,  $\hat{m}_2 = 2^{2n}(2^n + 1)$ ,  $\hat{m}_3 = 2^{2n}(2^n - 1)$ ,  $\left| \hat{m}_1^{-1} \right|_{m_1} = 2^{2n} - 1$ ,  $\left| \hat{m}_2^{-1} \right|_{m_2} = 2^{n-1}$  e  $\left| \hat{m}_3^{-1} \right|_{m_3} = 2^{n-1} + 1$ .

Eq.(1) pode se escrever como:

$$X = \sum_{i=1}^N \hat{m}_i \left| \hat{m}_i^{-1} \right|_{m_i} R_i - MA(X), \quad (2)$$

onde  $A(X)$  é um inteiro que depende do valor de  $X$ . Se fazemos uma divisão por  $m_1$  obtemos que Eq.(2) é:

$$X = \left\lfloor \frac{X}{m_1} \right\rfloor m_1 + R_1, \quad (3)$$

onde  $\left\lfloor \frac{X}{m_1} \right\rfloor$  está concadenado a  $R_1$ .

Para realizar a computação de  $\left\lfloor \frac{X}{m_1} \right\rfloor$ , dividimos Eq.2 por  $m_1$  e fazemos arredondamento por baixo fazendo modulo  $\hat{m}_1$ :

$$\begin{aligned} \left\lfloor \frac{X}{m_1} \right\rfloor &= \left\lfloor \left\lfloor \frac{X}{m_1} \right\rfloor \right\rfloor_{\hat{m}_1} = \left\lfloor \left\lfloor \sum_{i=1}^3 \left| \hat{m}_i^{-1} \right|_{m_i} \frac{\hat{m}_i}{m_1} R_i \right\rfloor_{\hat{m}_1} - \overbrace{\left\lfloor \frac{M}{m_1} A(X) \right\rfloor_{\hat{m}_1}}^{=0} = \\ &= \left\lfloor \left\lfloor \left| \hat{m}_1^{-1} \right|_{m_1} \frac{\hat{m}_1}{m_1} R_1 \right\rfloor_{\hat{m}_1} + \left\lfloor \overbrace{\left| \hat{m}_2^{-1} \right|_{m_2} \frac{\hat{m}_2}{m_1} R_2}^{V_2} \right\rfloor_{\hat{m}_1} + \left\lfloor \overbrace{\left| \hat{m}_3^{-1} \right|_{m_3} \frac{\hat{m}_3}{m_1} R_3}^{V_3} \right\rfloor_{\hat{m}_1} = \\ &= \left\lfloor \overbrace{(2^{2n} - 2)}^{V_1 = -1} R_1 + \overbrace{(2^{2n-1} + 2^{n-1})}^{V_2} R_2 + \overbrace{(2^{2n-1} - 2^{n-1})}^{V_3} R_3 \right\rfloor_{\hat{m}_1}. \end{aligned} \quad (4)$$

Podemos finalmente expressar os termos da Eq.4 para  $n = 4$  como:

$$\begin{aligned} V_1 R_1 &= \bar{r}_{1,7} \dots \bar{r}_{1,0}. \\ V_2 R_2 &= r_{2,0} \dots \\ V_3 R_3 &= \dots \end{aligned} \quad (5)$$

Devido a que alguns termos são números negativos, vamos precisar do factor corrector associados aos termos  $-R_1$  e  $-2^{n-1}R_3$  do mesmo modo que a aula anterior. Em este caso os factores correctores são escolhido como o mínimo valor que satisfaz:

$$\left| \text{abs} \left( \sum_{j=0}^{2n-1} |2^j|_{2^{2n-1}} \cdot \bar{r}_{1,j} \right) + COR_1 \right|_{2^{2n-1}} = 0. \quad (6)$$

$$\left| \text{abs} \left( \sum_{j=0}^n |2^{j+n-1}|_{2^{2n-1}} \cdot \bar{r}_{3,j} \right) + COR_3 \right|_{2^{2n-1}} = 0. \quad (7)$$

Os valores dos factores  $COR_1$  e  $COR_3$  são obtidos forçando as entradas  $R_1$  e  $R_3$  a  $\underbrace{(n-1)-bits}_{(n-1)-bits}$

0. O valor de  $COR_1 = 0$  e  $COR_3 = \underbrace{1 \dots 1}_{(n-1)-bits}$ , pelo que o valor de  $COR_{TOTAL} = \underbrace{1 \dots 1}_{(n-1)-bits}$ , corre ao cargo do aluno encaixar este factor corrector no VHDL dado usando unicamente 4 vectores  $A$ ,  $B$ ,  $C$  e  $D$ .

### 3 Implementação em VHDL do conversor RNS-Binario

- Com a placa em funcionamento, baixe o arquivo "Lab1b.zip" disponível no site da disciplina e descompacte esse arquivo na pasta `/Desktop/EEL51069/lab01b`. 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 12.1sp2 também pode ser utilizada). Com o software em funcionamento, acesse o menu File e a opção Open Project (File → Open Project) para abrir o projeto disponível na pasta destino da descompactação. Atenção: não use a opção "File → Open" para abrir o projeto, mas sim a "File → Open Project".
- Uma vez aberto o projeto, clique na entidade *traditionalSystem\_RNStoBin* disponível na aba Hierarchy do Project Navigator do Quartus II.
- Com o projeto e a entidade principal abertos, você deverá ver uma janela com a descrição em VHDL do conversor RNS-binario.

Nas linhas 1 → 16 estão definidas as livrerias, as entradas e saídas da estrutura estão definidas nas linhas 18 → 22 e a arquitetura a partir da linha 26.

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

Agora o aluno deve preencher partes do código VHDL para assim obter um conversor RNS-Binario usando o conjunto de modulos  $\{2^{2n}, 2^n - 1, 2^n + 1\}$  e  $n = 4$ . Nota: O aluno tem de tem em consideração que as entradas do circuito  $R_1 = \{r_{1,(2n-1)}, \dots, r_{1,1}, r_{1,0}\}$ ,

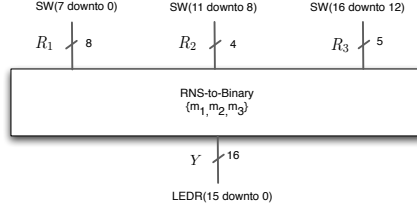


Figura 2: Bloco RNS-binário com associação de pinos entrada-saída.

Tabela 1: Tabela de resultados de simulação

Canal $m_1$	Canal $m_2$	Canal $m_3$	$X$
0	0	0	
255	1	1	
252	0	0	
162	10	5	

$R_2 = \{r_{2,(n-1)}, \dots, r_{2,1}, r_{2,0}\}$ , e  $R_3 = \{r_{3,(n)}, \dots, r_{3,1}, r_{3,0}\}$  estão associadas aos Swit-ches 7 a 0,  $R_1$ , Switches 11 a 8,  $R_2$ , Switches 16 a 12,  $R_3$ , e as saídas estão associadas aos LEDs vermelhos 15 a 0 como é mostrado na Fig. 2.

A Fig. 3 mostra o diagrama de blocos para a implementação da Eq.3.

**Para a implementação da soma dos termos  $V_i R_i$ ,  $1 \leq i \leq 3$  usaremos um CSA que soma os três termos fornecendo os arrays Acarreio  $C = c_n, \dots, c_2, c_1$  e Soma  $S = s_{n-1}, \dots, s_1, s_0$ .** Tendo em consideração que  $|2^{2n} c_n|_{2^{2n-1}} = c_n$ , podemos recolocar o bit  $c_n$  na posição  $2^0$  (i.e: operação End-Around-Carry). No VHDL os arrays de multiplicação de  $V_1 R_1$ ,  $V_2 R_2$ ,  $V_3 R_3$  fornecem 4 vectores  $A$ ,  $B$ ,  $C$  e  $D$ .

A continuação devem ser somadas nas seguintes etapas o termo  $-2^{n-1} R_3$  e o factor corrector  $COR$ . **O aluno deve implementar em VHDL estas somas.**

Finalmente a soma na ultima etapa é feita por um Carry Propagate Adder (CPA) também com EAC. O CSA+EAC é implementado no VDHL *CSA\_2n\_mp\_1.vhd* previamente fornecido usando o bit de controle *modo* = 0. **Preencha o resto do VHDL linhas 65  $\rightarrow$  83 usando os sinais auxiliares dados nas linhas 52 a 57.**

A saída do CPA com EAC é o valor de  $\left\lfloor \frac{X}{m_1} \right\rfloor$  da Eq.4 o qual deve ser concatenado com  $R_1$  como mostrado na Eq.3. **Preencha o resto do VHDL para realizar dita concatenação.**

Uma vez preenchido o VHDL compile o projeto até não ter erros na descrição. Uma vez compilado abra modelsim e simule o circuito. Dica: use o script .do para forçar as entradas. **Preencha a tabela 1 com os dados da simulação.**

Uma vez terminada a simulação implemente na placa DE2 o circuito e **preencha a tabela 2 com os resultados obtidos nos LEDs vermelhos.**

**A continuação copie o arquivo "traditionalSystem\_BintoRNS.vhd" obtido no laboratório 1a ao projeto atual. Ligue as saídas de dito circuito às**

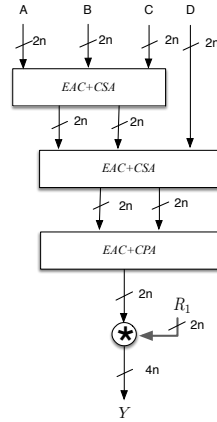


Figura 3: Diagrama de blocos para conversor RNS-binário para o módulo  $\{m_1, m_2, m_3\} = \{2^{2n}, 2^n - 1, 2^n + 1\}$ .

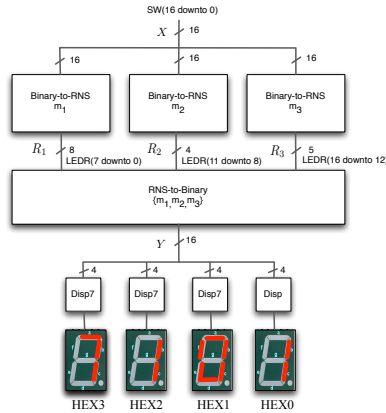


Figura 4: Diagrama de blocos para conversor Binário-RNS-binário para o módulo  $\{m_1, m_2, m_3\} = \{2^{2n}, 2^n - 1, 2^n + 1\}$ .

entradas do "traditionalSystem\_RNStoBin.vhd". Agora interligue a saída de "traditionalSystem\_RNStoBin.vhd" aos Displays HEX3, HEX2, HEX1, HEX0 como mostrado na Fig.4 usando o Disp7seg.vhd dado no site da disciplina. Interprete os resultados obtidos.

Importante: antes de desligar o computador, guarde a pasta lab1b em um pendrive ou envie por e-mail já que este circuito será usado nos seguintes laboratórios.

Tabela 2: Tabela de resultados na placa DE2

<b>Canal</b> $m_1$	<b>Canal</b> $m_2$	<b>Canal</b> $m_3$	$X$
0	0	0	
255	1	1	
252	0	0	
162	10	5	

### 3.2 Questões finais

- **Pergunta 1:** Explique como encaixou o  $COR_{TOTAL}$  sem a necessidade de adicionar mais um nível de  $CSA + EAC$ .