UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMATICA
Aula 26/1/23 - Sistemas Digitais – Parte 2
Prof. Fernanda Kastensmit

Objetivo do Trabalho 1 a ser entregue em Fevereiro 10/2: projetar e descrever em VHDL o processador Ahmes, implementar 2 programas em sua memória e mostrar através de simulação lógica sem e com atraso o funcionamento.

Parte 2: referente a aula **26/1/2022**

Descrever o programa no Ahmes de multiplicação de dois números inteiros positivos de 8 bits por Deslocamento e soma em binário e colocar no arquivo .COE na memória BRAM.

Cole aqui o programa em ASSEMBLY e o .COE

```
;Programa 1 - Flag tester
   ; Programa para gerar diferentes flags
   ;no processador didatico Ahmes para testar
   ;com a implementaçao desse em VHDL

; inicializa na segunda posiçao da memoria
ORG 1
LDA cte_128  ; gera flag N
LDA cte_0 ; gera flag Z
LDA cte_127
ADD cte_1  ; gera flags N e V
LDA cte_255
ADD cte_1 ; gera flags Z e C
LDA cte_1
SUB cte_255 ; gera flags C e B
HLT

ORG 128
cte_0:
        DB 0
cte_1:
        DB 1
cte_127:
        DB 127
cte_128:
        DB 128
cte_254:
        DB 254
cte_255:
        DB 255
```

Arquivo .coe:

```
memory_initialization_radix=10;
memory_initialization_vector=0,32,131,32,128,32,130,48,129,32,133,48,129,32,129,112,1
33,240,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,127,128,254,255,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0;
```

```
;Nome: Giordano Souza
;Cartão UFRGS: 00308054

;Programa 2 - Instruction tester
   ; Programa para testar as diferentes
   ;intruções do processador didatico Ahmes
   ;em sua implementaçao em VHDL

; inicializa na segunda posiçao da memoria
ORG 1
  ;Flags se referem a N Z V C B respectivamente
NOP                 ;AC -> 0   Flags: 0 1 0 0 0
STA dummy           ;AC -> 0   Flags: 0 1 0 0 0
; end 128 tinha 10 - passa a ter 0
LDA cte_1           ;AC -> 1   Flags: 0 0 0 0 0
ADD cte_1           ;AC -> 2   Flags: 0 0 0 0 0
OR cte_127          ;AC -> 127 Flags: 0 0 0 0 0
AND cte_64          ;AC -> 64  Flags: 0 0 0 0 0
NOT                 ;AC -> 191 Flags: 1 0 0 0 0
SUB cte_1           ;AC -> 190 Flags: 1 0 0 0 0
JMP label_0         ;AC -> 190 Flags: 1 0 0 0 0

label_0:
    JN label_1      ;AC -> 190 Flags: 1 0 0 0 0
label_1:
  JP label_incorreto  ;AC -> 190 Flags: 1 0 0 0 0
  JV label_incorreto  ;AC -> 190 Flags: 1 0 0 0 0
  JNZ label_2         ;AC -> 190 Flags: 1 0 0 0 0
label_2:
  JC label_incorreto  ;AC -> 190 Flags: 1 0 0 0 0
  JNC label_3         ;AC -> 190 Flags: 1 0 0 0 0
  JMP label_incorreto ; nao deve executar
label_3:
  JB label_incorreto  ; nao deve executar
  JNB label_4         ;AC -> 190 Flags: 1 0 0 0 0
  JMP label_incorreto ; nao deve executar
label_4:
  SHR                 ;AC -> 95  Flags: 0 0 0 0 0
  SHL                 ;AC -> 190 Flags: 1 0 0 0 0
```

```
  ROR             ;AC -> 95  Flags: 0 0 0 0 0
  ROL             ;AC -> 190 Flags: 1 0 0 0 0
  HLT  ; para o programa
label_incorreto:
        ; se o programa chegar nesse halt
        ; algo de errado aconteceu
        HLT ; para o programa

;STA

ORG 128
dummy:
        DB 10
cte_0:
        DB 0
cte_1:
        DB 1
cte_64:
        DB 64
cte_127:
        DB 127
cte_128:
        DB 128
cte_254:
        DB 254
cte_255:
        DB 255
```

Arquivo .coe :

```
memory_initialization_radix=10;
memory_initialization_vector=0,0,16,128,32,130,48,130,64,132,80,131,96,112,130,128,17
,144,19,148,42,152,42,164,25,176,42,180,31,128,42,184,42,188,37,128,42,224,225,226,2
27,240,240,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,64,
127,128,254,255,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0;
```

*****ENTREGAR da PARTE 2 DIA 29/1/2023**** Editando esse DOC e submetendo no MS-TEAMS**