

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
ENGENHARIA DE COMPUTAÇÃO

ADALBERTO TEIXEIRA GUEDES
ANDERSON SOARES DA SILVA
GUIDO MARGONAR MOREIRA
GUILHERME YAMAMOTO KATO
GUSTAVO CARVALHO ALICRIM DA SILVA

TRABALHO 1
PROCESSADOR

APUCARANA
2021

Turma: Circuitos Digitais - 2021.1 - COM2D

--DOCUMENTACAO--

--MANUAL OPCODE--

--SOMA	- COD	-	'0000'
--SUBTRACAO	- COD	-	'0001'
--MULTIPLICACAO	- COD	-	'0010'
--DIVISAO	- COD	-	'0011'
--IGUALDADE	- COD	-	'0100'
--MAIOR QUE	- COD	-	'0101'
--MENOR QUE	- COD	-	'0110'
--MAIOR OU IGUAL	- COD	-	'0111'
--MENOR OU IGUAL	- COD	-	'1000'
--DIFERENTE	- COD	-	'1001'

--MANUAL OPCODE EXTRA--

--OP. MOVE DO REGISTRADOR A PARA O B	- COD	-	'1010'
--OP. MOVE DO REGISTRADOR B PARA O A	- COD	-	'1011'
--OP. LOAD Da MEMORIA PARA O REGISTRADOR	- COD	-	'1100'
--OP. STORE Do REGISTRADOR PARA A MEMORIA	- COD	-	'1101'

--SWITCHES--

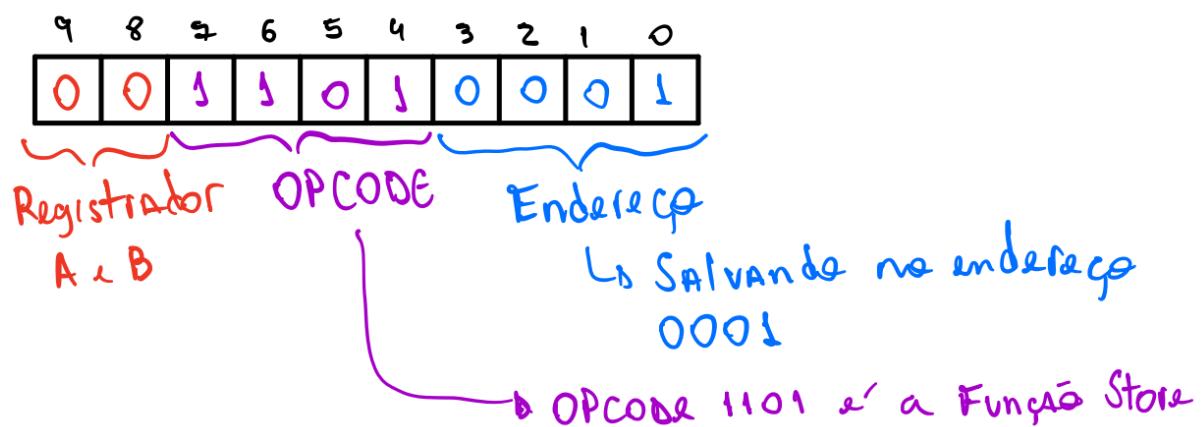
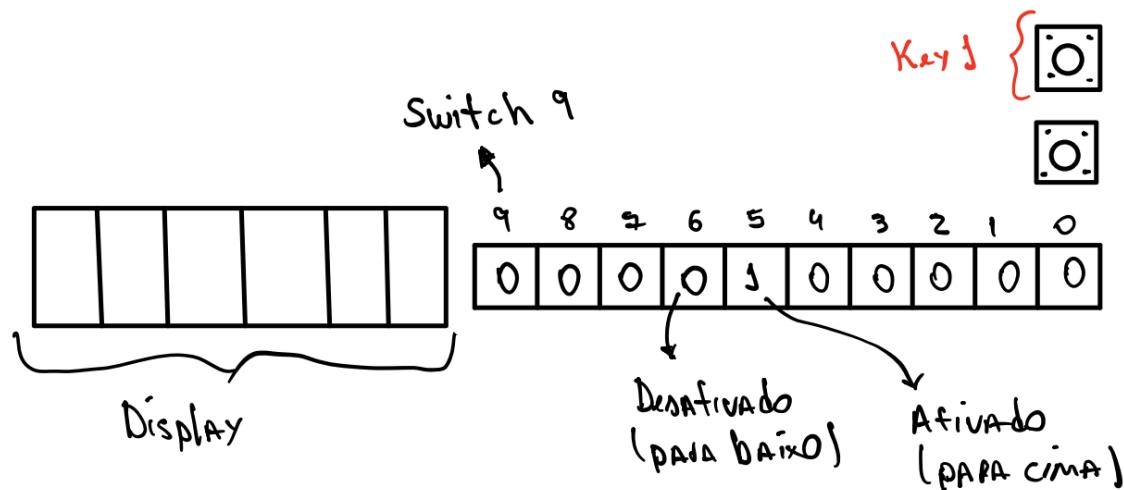
--NAME	- PINO	-	FUNÇÃO
--SW0	- PIN_C10	-	ENDEREÇO DE MEMORIA
--SW1	- PIN_C11	-	ENDEREÇO DE MEMORIA
--SW2	- PIN_D12	-	ENDEREÇO DE MEMORIA
--SW3	- PIN_C12	-	ENDEREÇO DE MEMORIA
--SW4	- PIN_A12	-	OPCODE
--SW5	- PIN_B12	-	OPCODE
--SW6	- PIN_A13	-	OPCODE
--SW7	- PIN_A14	-	OPCODE
--SW8	- PIN_B14	-	SELETOR DE REGISTRADOR
--SW9	- PIN_F15	-	SELETOR DE REGISTRADOR

--BOTOES--

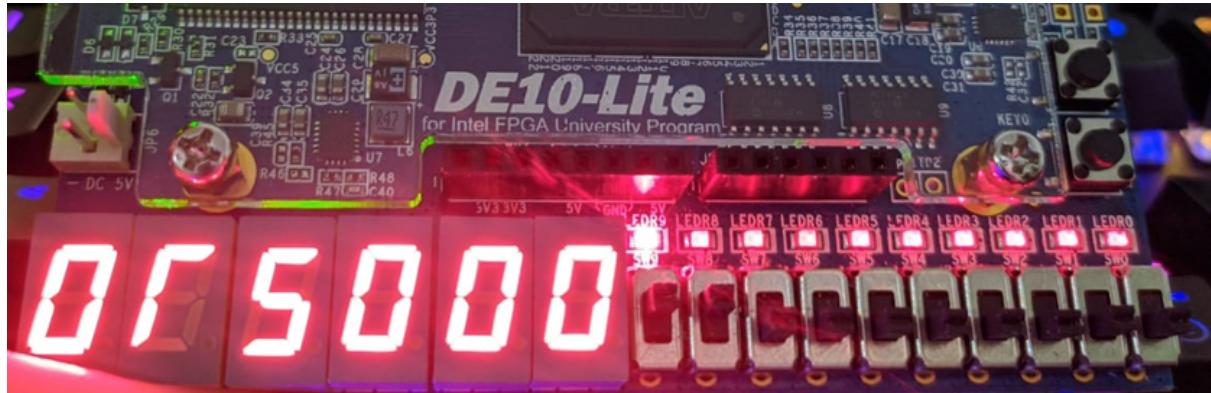
--NAME	- PINO	-	FUNÇÃO
--KEY1	- PIN_A7	-	HABILITAR IMPUT DE DADOS NO REGISTRADOR

--LEDS--

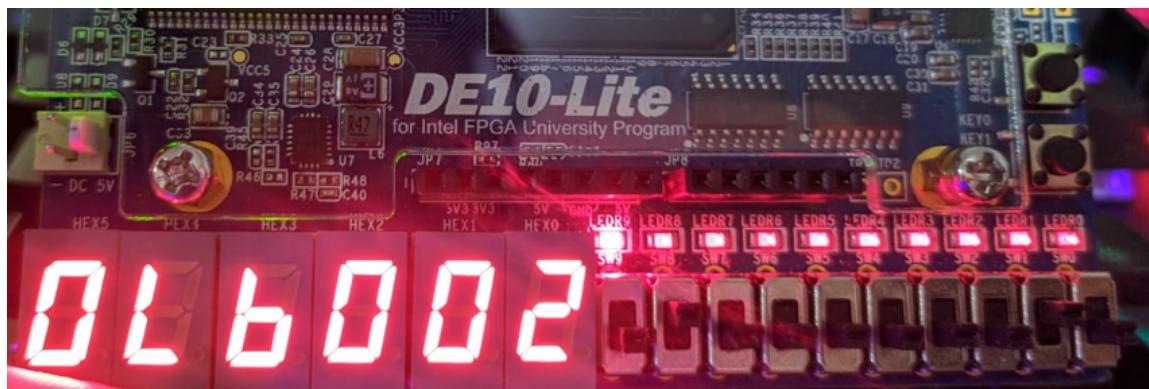
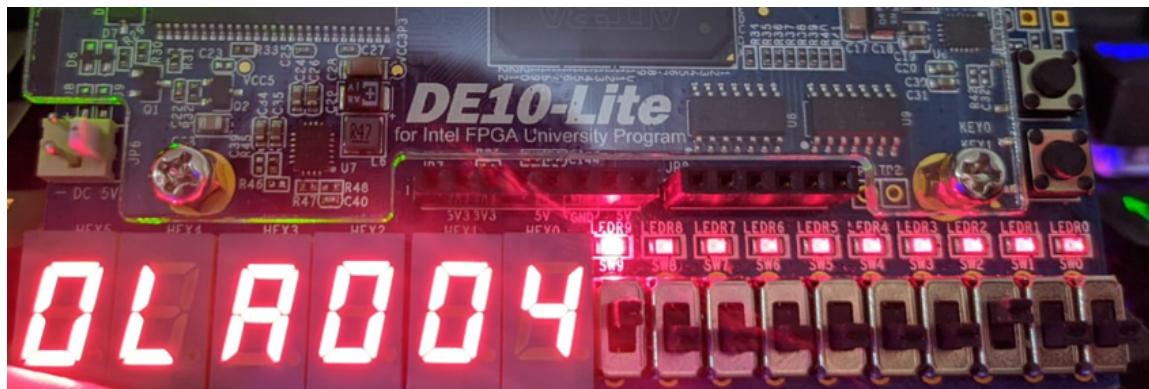
--NAME	- PINO	-	FUNÇÃO
--LEDR0	- PIN_A8	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR1	- PIN_A9	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR2	- PIN_A10	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR3	- PIN_B10	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR4	- PIN_D13	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR5	- PIN_C13	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR6	- PIN_E14	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR7	- PIN_D14	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR8	- PIN_A11	-	RESULTADO DA OPERAÇÃO PROPOSTA
--LEDR9	- PIN_B11	-	RESULTADO DA OPERAÇÃO PROPOSTA



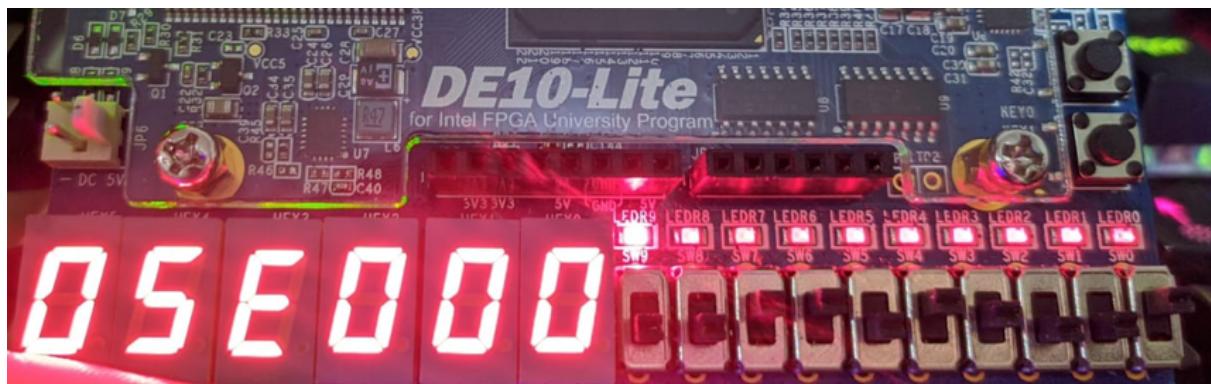
Usando o **Reset**, que ocorre quando os dois primeiros switch da esquerda (switch 8 e 9) estão levantados e o botão **key1 (CLOCK)** é pressionado (**1100000000**)



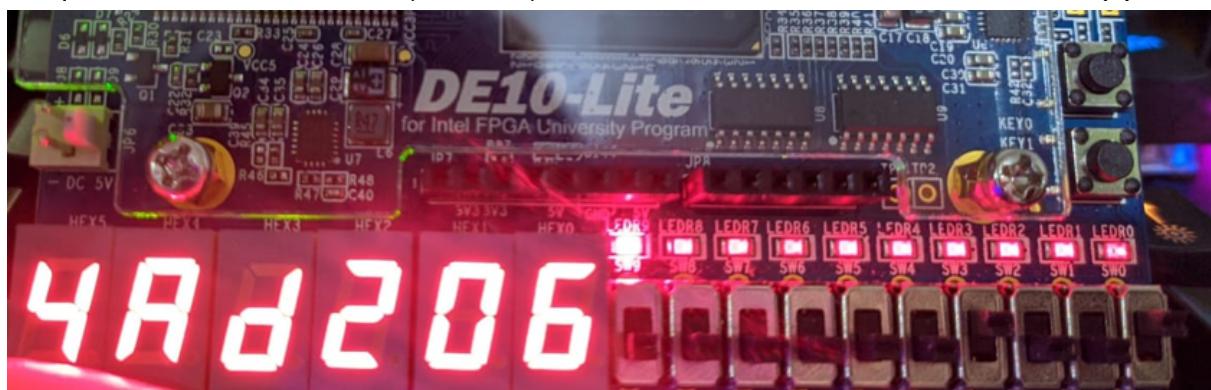
Usando o **Registrador A** para escolher o primeiro número binário, quando o switch 9 (**1000000000**) é levantado, os switch 7 ao 0 são usados para colocar os números em binário e se o key1 (**CLOCK**) for pressionado, o número é salvo no registrador. Na imagem abaixo está o número binário **00000100 (Decimal 4)**. O mesmo serve para o **registrador B** para escolher o segundo número binário que no caso é **00000010 (Decimal 2)**, mas o switch que precisa ser levantado é o switch 8 (**0100000000**).



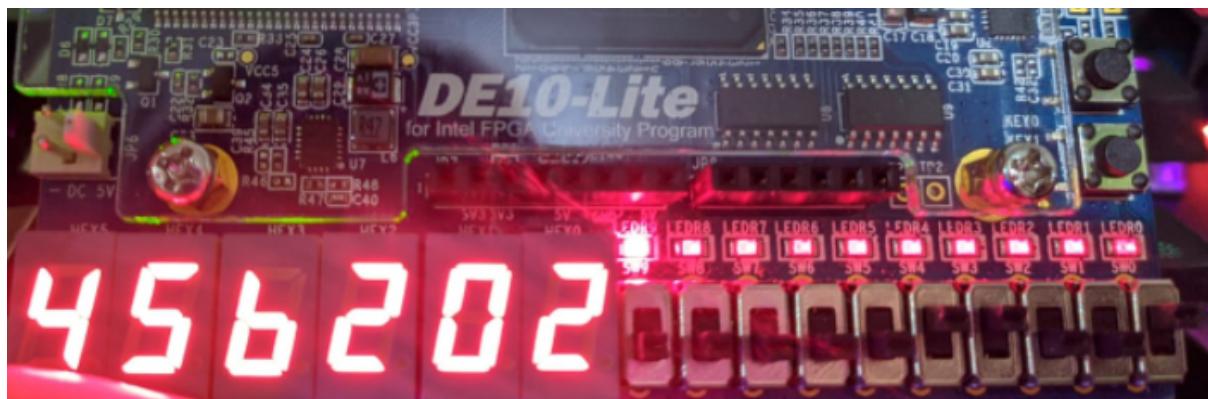
Depois utilizar o **Store (OPCODE 1101)** para armazenar os números dos registradores A e B na memória quando key1 (**CLOCK**) for pressionado. Para escolher o endereço para ser salvo na memória, utilize os switch 0 ao 3, no exemplo foi salvo no endereço **1001**.



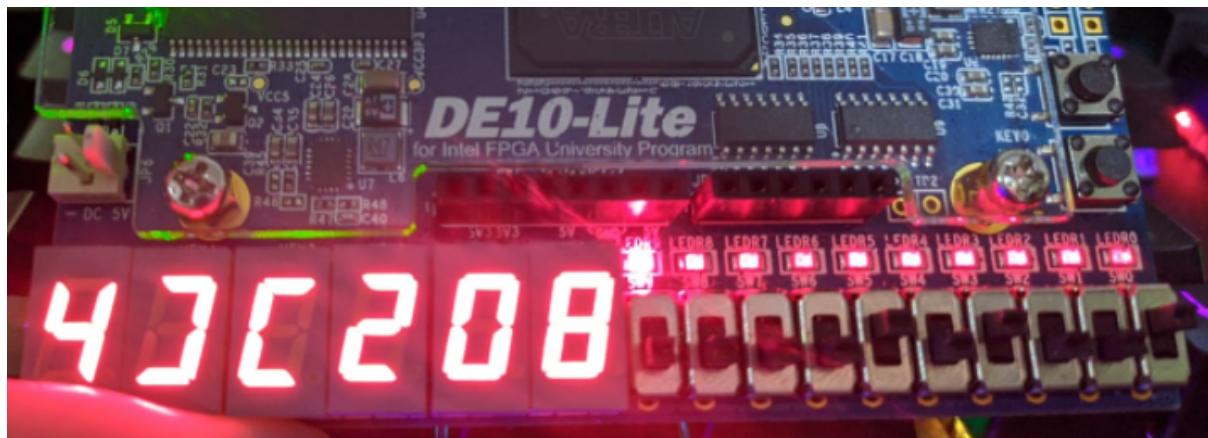
Com os números binários salvos na memória sendo o **00000100** e **00000010**, conseguimos ver que com o **OPCODE 0000 (Adição)**, o resultado é o número binário **00000110 (6)**



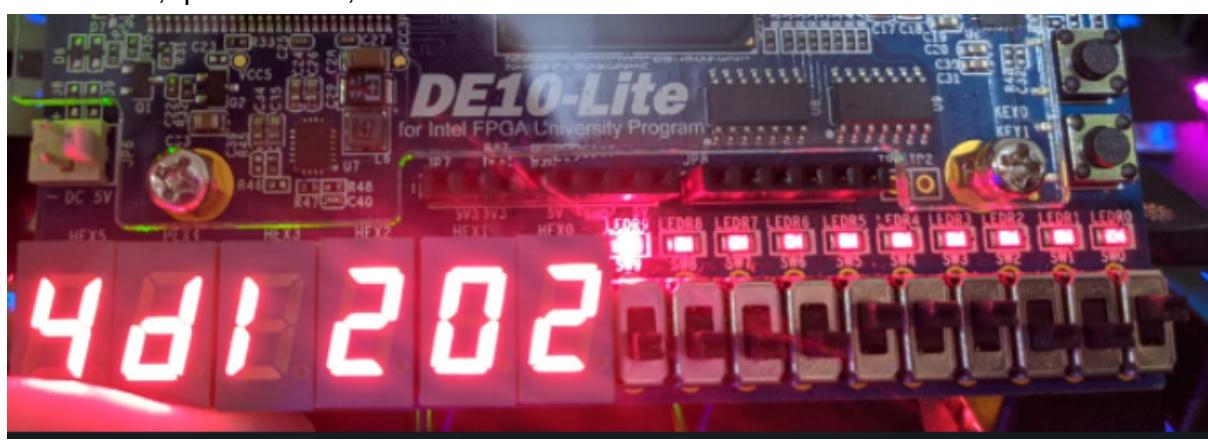
Utilizando o **OPCODE 0001 (Subtração)** podemos observar que com os números decimais sendo 4 e 2, o resultado é o número decimal 2



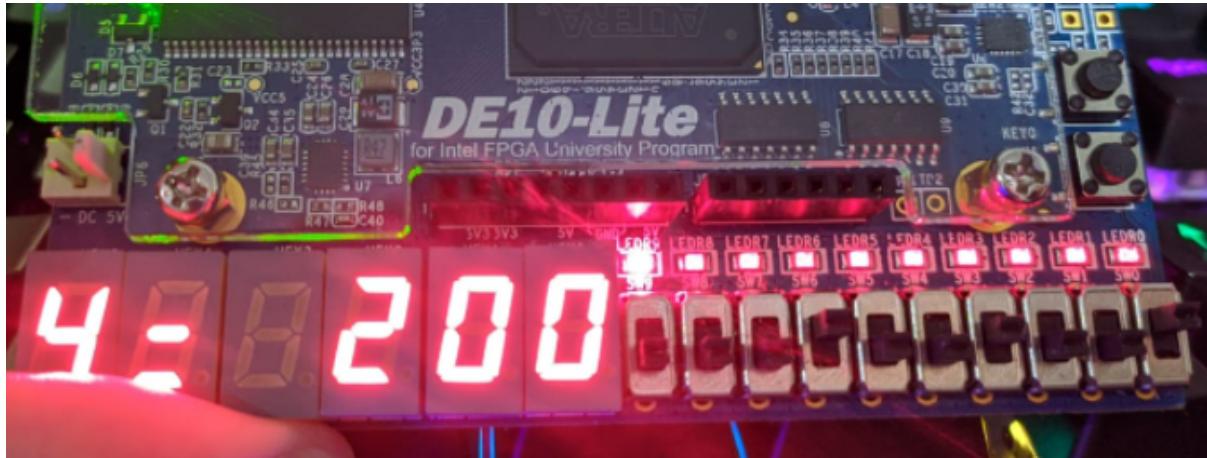
Utilizando o **OPCODE 0010 (Multiplicação)**, observamos que temos os números decimais 4 e 2 da memória, que multiplicando, obtemos o número 8.



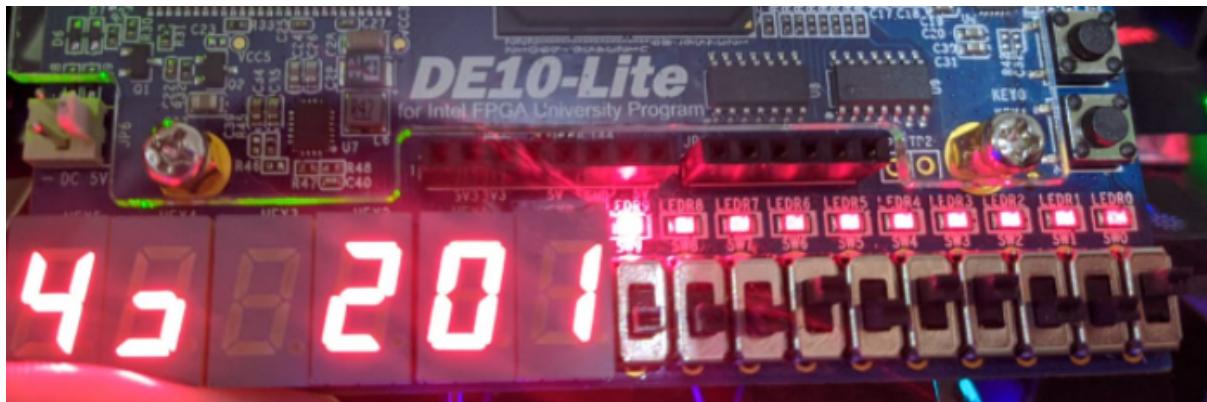
Utilizando o **OPCODE 0011 (Divisão)**, observamos que temos os números decimais 4 e 2 da memória, que dividindo, obtemos o número 2.



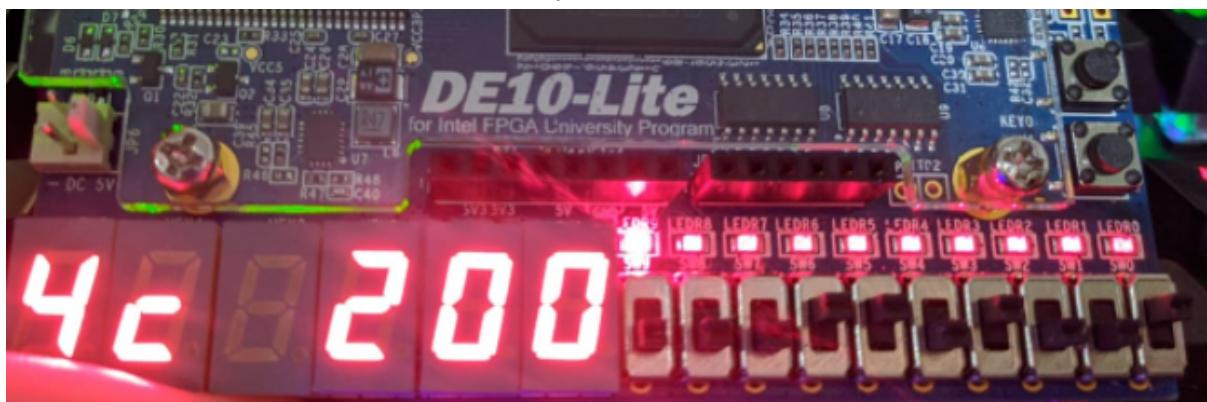
Utilizando o **OPCODE 0100 (Igualdade)**, observamos que temos os números decimais 4 e 2 da memória, e compararmos se os dois números são iguais, que no caso, não são, pelo último display mostrando o 0 (False).



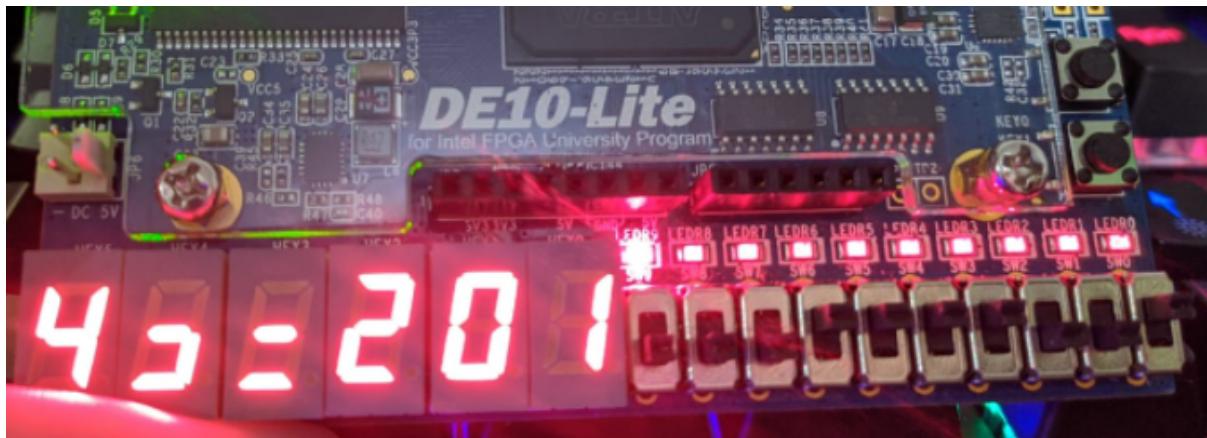
Utilizando o **OPCODE 0101 (Maior que)**, observamos que temos os números decimais 4 e 2 da memória, e compararmos se o primeiro número na memória é maior que o segundo, que no caso, ele é verdade, pelo último display mostrando o **1 (True)**.



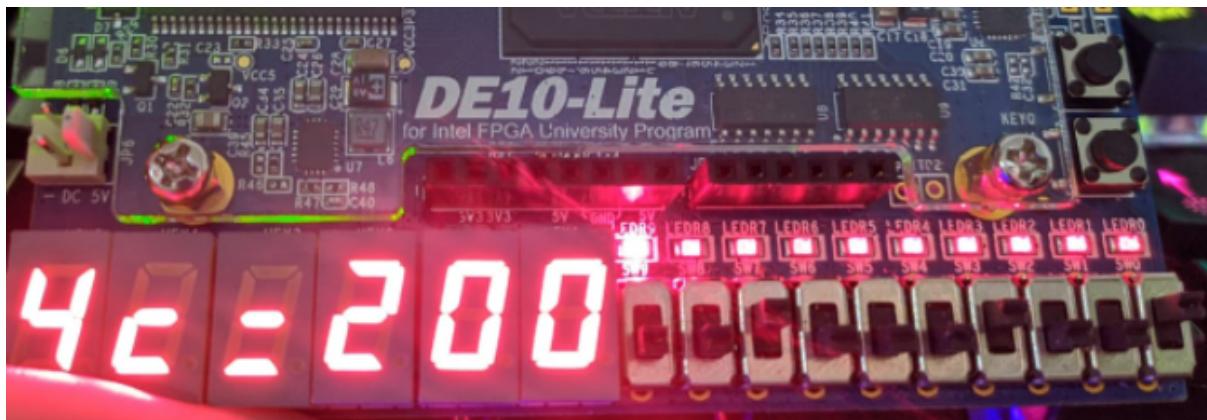
Utilizando o **OPCODE 0110 (Menor que)**, observamos que temos os números decimais 4 e 2 da memória, e compararmos se o primeiro número na memória é menor que o segundo, que no caso, ele é falso, pelo último display mostrando o **0 (False)**.



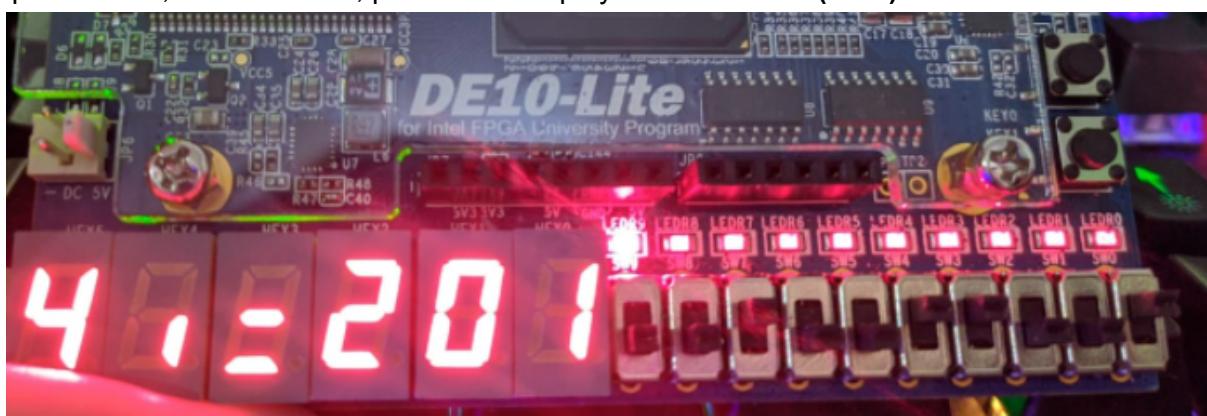
Utilizando o **OPCODE 0111 (Maior ou igual)**, observamos que temos os números decimais 4 e 2 da memória, e compararmos se o primeiro número na memória é maior ou igual ao segundo, que no caso, ele é verdade, pelo último display mostrando o **1 (True)**.



Utilizando o **OPCODE 1000 (Menor ou igual)**, observamos que temos os números decimais 4 e 2 da memória, e compararmos se o primeiro número na memória é menor ou igual ao segundo, que no caso, ele é falso, pelo último display mostrando o **0 (False)**.

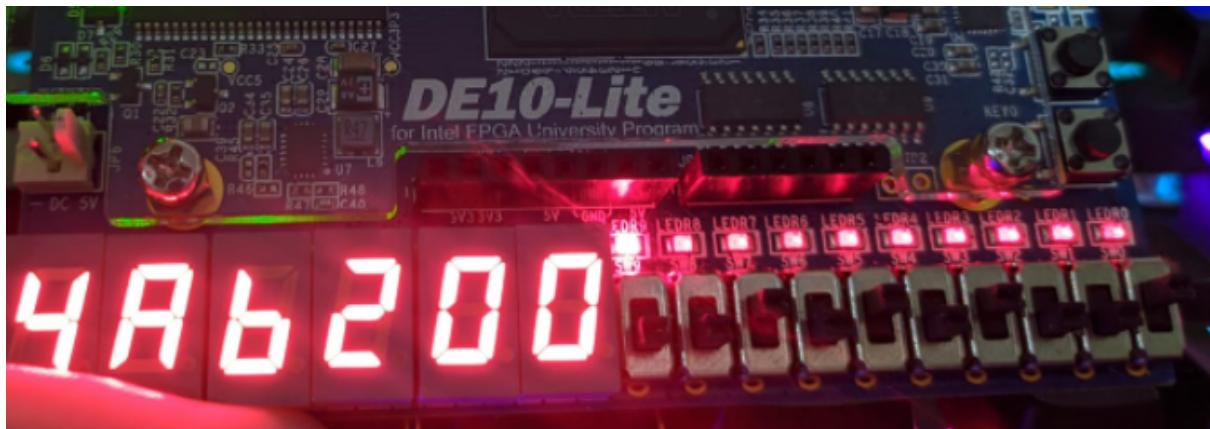


Utilizando o **OPCODE 1001 (Diferente)**, observamos que temos os números decimais 4 e 2 da memória, e compararmos se o primeiro número na memória é diferente ao segundo, que no caso, ele é verdade, pelo último display mostrando o **1 (True)**.



Utilizando o **OPCODE 1010 (Registrador A para registrador B)**, observamos que temos os números decimais 4 e 2 da memória, e for pressionado o **key1 (CLOCK)** , o número 4 do

registrador A, vai ser copiado para o registrador B, ficando com o número 4 no registrador A e B.



Utilizando o **OPCODE 1100**, os números binários da memória atual que no caso do exemplo é o **endereço 1001**, é salva nos **registradores A e B**

