

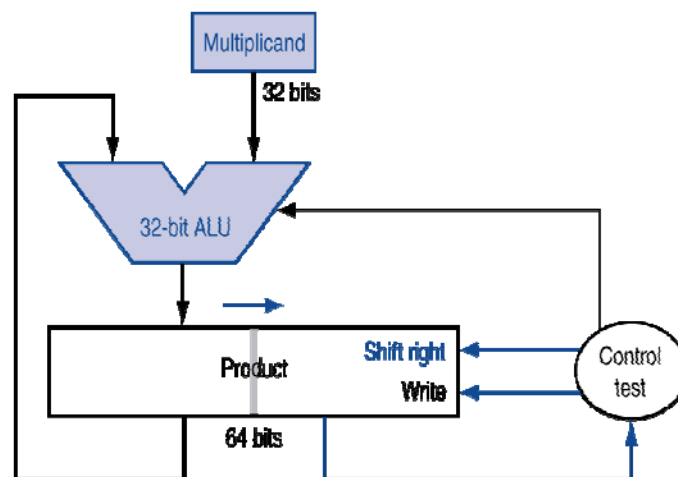
Exercícios 03

1. Considerando o loop escrito em linguagem de alto nível:

```
while (save[i] == k)
    i = i + j;
```

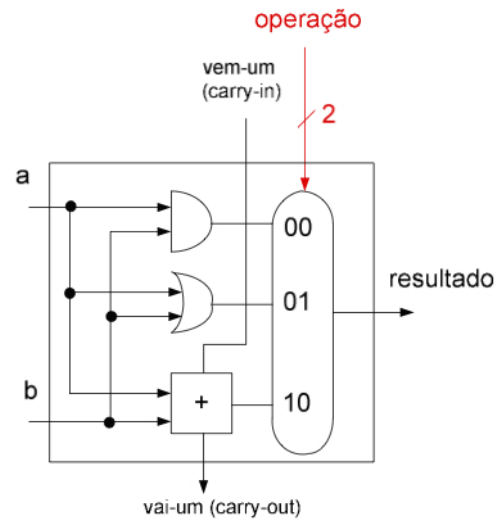
e supondo que i , j e k correspondam às variáveis $\$s3$, $\$s4$ e $\$s5$, e que o endereço inicial do array *save* esteja armazenado em $\$s6$, qual o código em linguagem de montagem para a arquitetura MIPS correspondente a este trecho de código?

2. Os números abaixo se encontram na representação de Complemento de Dois. Efetue as seguintes operações:
- $111000 - 110011$
 - $11001100 - 101110$
 - $01101111 + 01100010$
 - $111100001111 - 110011110011$
 - $11000011 - 11101000$
 - $11010010 - 01111101$
3. Supondo que o registrador $\$t4$ da ISA MIPS contém a constante 100, gere um trecho de código MIPS que multiplique a constante anterior por um valor armazenado no registrador $\$t2$ e verifique a ocorrência de overflow.
4. Utilizando a representação do hardware de multiplicação ilustrado abaixo, apresente uma tabela contendo as iterações e os passos realizados em cada iteração da multiplicação dos números de 4 bits 0110_2 e 0011_2



5. Escrever um programa em Assembly do MIPS que imprime os 10 primeiros elementos da série de Fibonacci (0, 1, 1, 2, 3, 5 ...).

6. Dado o circuito da ULA de 1 bit abaixo, como seria possível selecionar a execução de uma operação XOR entre a e b, sem alterar o circuito? Nota-se que essa operação XOR é apenas para a ULA de 1 bit, portanto não precisa funcionar quando se encadeia 32 ULAs de 1 bit.



7. A operação slt resulta em 1 se $a < b$ e 0, caso contrário. Como seria possível construir uma ULA com uma operação slt modificada, em que resulte em -1 se $a < b$ e 0, caso contrário?