**Multiplicador:**

**Primeira Versão:**

* Resultado: 64 bits
* Multiplicador: 32 bits
* Multiplicando: 64 bits
* ALU: 64 bits.

A close up of text on a white background

Description automatically generated

A screenshot of a cell phone

Description automatically generated

Exemplo para 4 bits:

A close up of a keyboard

Description automatically generated

Shift do multiplicando (a vermelho, para a esquerda), seguido da soma (com, o resultado, anterior).

**Segunda Versão:**

* Deslocar o resultado para a direita e não o multiplicando para a esquerda
* Multiplicador continua a ser deslocado para a direita
* ALU: 32 bits.
* Multiplicando: 32 bits
* Multiplicador: 32 bits
* Resultado: 64 bits

A screenshot of a cell phone

Description automatically generated

No inicio, o resultado é tudo a zeros, entao podemos guardar o multiplicador nos LSB do resultado porque ambos vao dar shift right, entao vamos utilizar sempre o LSB do resultado que é o LSB do multiplicador. até ele desaparecer e ficar só o resultad.

A close up of a keyboard

Description automatically generatedexemplo:

0011

x 0101

---------------

resultado na iteracao 0/1:

inicio: 0000 0101

add: 0011 0101

com sift: 0001 1010

**Multiplicação de inteiros com sinal:**

* Para a arquitectura de multiplicação de números com sinal, utiliza-se o algoritmo de Booth.