

SSC0511 - Organização de Computadores Digitais 1

Prof. Fernando Santos Osório
Monitor: André Bannwart Perina

Lista 2 - Processador 6502

Nota: Para exercícios de impressão em tela, o simulador do Kowalski disponibiliza uma interface de I/O. Para utilizá-la, basta colar o código abaixo no início de seu código e utilizar os identificadores com as instruções LD* e ST*:

```
1 ; Enderecos do terminal
2 io_clr = $E000 ; Limpa terminal
3 io_putc = $E001 ; O codigo ASCII sera enviado ao terminal e mostrado como um caractere
4 io_ptci = $E002 ; O codigo ASCII sera enviado ao terminal e mostrado como um caractere ,
   mas CR/LF e ignorado
5 io_puth = $E003 ; O codigo ASCII sera enviado ao terminal e mostrado como um numero
   hexadecimal
6 io_getc = $E004 ; Se um caractere e escrito no terminal, este sera escrito para esta
   localizacao
7 io_setx = $E005 ; Controla localizacao horizontal do cursor
8 io_sety = $E006 ; Controla localizacao vertical do cursor
```

1. Com base no algoritmo de soma de 16 bits passado em aula, desenvolva um algoritmo para cálculo de soma entre dois positivos de n bytes, utilizando loops.
2. Faça um programa para imprimir “Hello World!” na tela. Ao declarar a string, declare o último caractere como ASCII 0 (NULL) e utilize-o para terminar a rotina (como ocorre em C).
3. Utilizando o algoritmo desenvolvido no exercício 1 ou em aula, faça uma subrotina para imprimir a operação realizada em tela, como se segue (**Dica:** Use `io_puth` para imprimir os bytes em hexadecimal):

```
1   0002FF
2   +E100F4
3   -----
4   E103F3
```

4. Faça uma pequena implementação de `printf`. Ou seja, implemente uma subrotina de impressão de strings utilizando chamada de função com JSR e RTS. Antes de chamar a função, o programa deve copiar a string (terminada com NULL) para uma área reservada e especificar a posição X e Y para a impressão (também em variáveis predefinidas). Não é necessário tratar quebra de linha ou qualquer outra funcionalidade do `printf`. Utilize as variáveis abaixo como parâmetros entre o programa principal e a função::

```
1 posx: .DB $0 ; Posicao X para impressao
2 posy: .DB $0 ; Posicao Y para impressao
3 strp: .RS $100 ; String a ser impressa. Espaco alocado de 256 bytes
```

5. Faça um programa para imprimir um número de 1 byte em binário (**Dica:** Dado v o valor a ser impresso, faça $v \& 0x1$ para pegar apenas o primeiro bit. Utilize LSR para realizar shift e fazer o mesmo com os bits mais significativos).