

Sistemas de Microprocessadores

DEP. DE ENG.^a ELECTROTÉCNICA E DE COMPUTADORES
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA



Lab 6 – Acesso a Periféricos

Neste trabalho de laboratório pretende-se programar o acesso a periféricos

Os exercícios que se seguem são para ser executados com o MARS que é um simulador do MIPS disponível em <http://courses.missouristate.edu/KenVollmar/MARS/>. Leia atentamente a documentação e explore a sua utilização usando o exemplo fibonacci.asm disponível no mesmo sítio.

Na aula deve explicar todos os detalhes dos exercícios realizados.

1. Display de 7 segmentos

O processador MIPS não tem instruções específicas para aceder a periféricos (ao contrário de processadores como os da família Intel x86). Assim os periféricos são mapeados no espaço de endereçamento normal. Ou seja, no caso do "Digital Lab Sim", acessível no menu Tools, existem 3 periféricos: 2 displays de 7 segmentos e um teclado hexadecimal.

Para aceder ao display da direita basta escrever um byte no endereço 0xFFFF0010. Note que cada bit do byte escrito corresponde a um dos elementos (barras e ponto) desse display.

- Faça um pequeno programa que num ciclo vá enviando bytes com apenas 1 bit a 1 para o display direito. Faça uso da instrução shift left logical (sll) para começando com 1 ir deslocando o bit a 1 para a esquerda. Execute o programa passo a passo e verifique qual o bit que corresponde a cada um dos segmentos anotando numa tabela.
- Faça uma tabela onde escreva os valores a enviar para o display para que este mostre os números de 0 a F (hexadecimal).
- Escreva um programa que vá contando de 0 até 255 e mostre nos dois displays o correspondente número em hexadecimal.
- (Opcional) Modifique o programa anterior para mostrar contagens de 0 a 99 em decimal.

2. Leitura do teclado por varrimento

O teclado simula um teclado normal em que as teclas são lidas através de um mecanismo de "scan". Isto significa que temos de ir testando linha a linha do teclado para ver se há alguma tecla premida. Assim para ler o teclado teremos de em ciclo ir enviando para o endereço 0xFFFF0012 o número da linha a testar e de seguida vamos ler no endereço 0xFFFF0014 se há alguma tecla premida nessa linha. O resultado lido se houver tecla premida será o composto por o número da linha (nibble mais significativo) e o número da

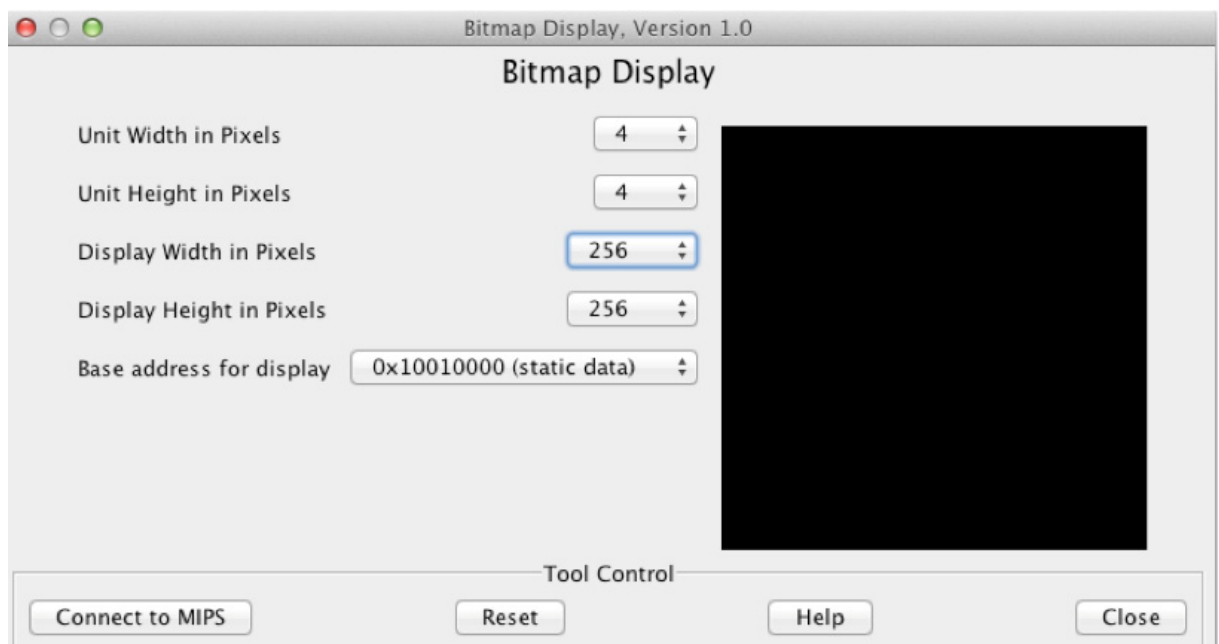
coluna (nibble menos significativo).

Faça um programa que use as funções desenvolvidas na questão anterior, esteja continuamente a testar o teclado e se houver uma tecla premida mostre no display direito o valor hexadecimal correspondente. Caso não haja tecla premida o display deve ficar apagado.

3. Bitmap Display

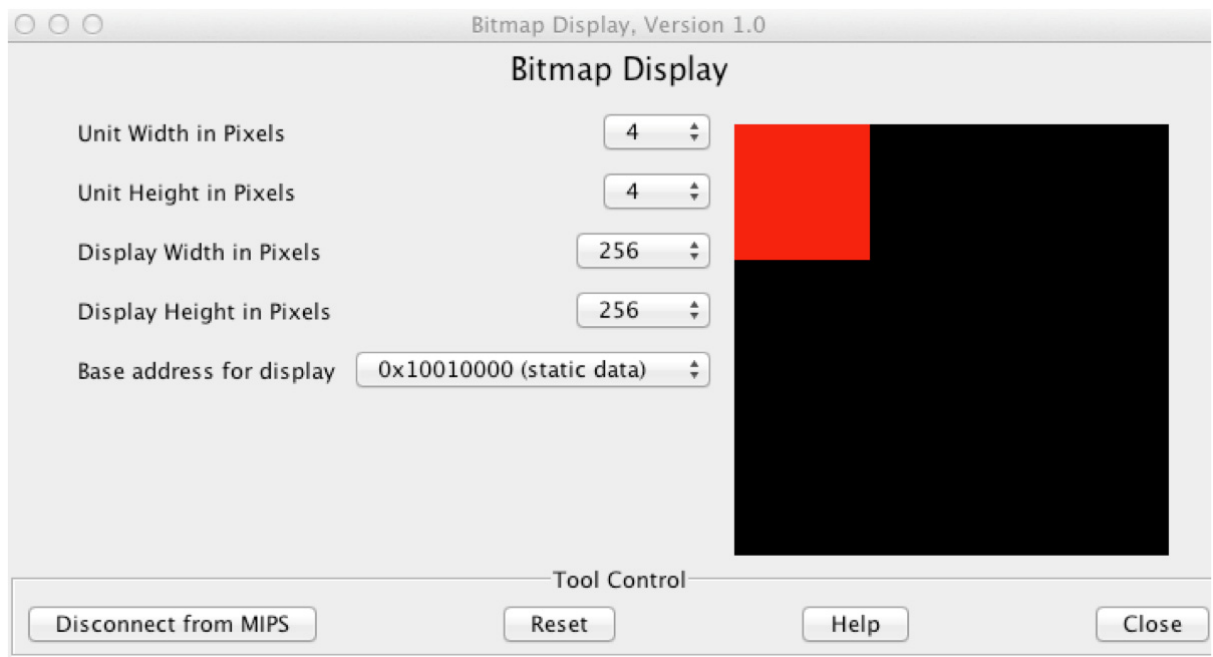
No menu Tools encontra-se disponível uma ferramenta que simula um display gráfico onde a cor de cada pixel corresponde à combinação de valores de vermelho, verde e azul armazenados numa palavra de 32 bits. Assim cada cor é definida pelos valores de vermelho (bits 16-23), verde (bits 8-15) e azul (bits 0-7), os bits 24 a 31 não são utilizados. Por exemplo para atribuir a cor amarela a um pixel deve-se escrever na zona de memória correspondente o valor 0x0000ffff.

Vamos escolher um display de 64 por 64 pixels, mas por razões de melhor visualização vamos seleccionar que cada um desses pixéis corresponde a um bloco de pixéis 4x4 do nosso ecrã. Ou seja configure a ferramenta de acordo com a figura seguinte.

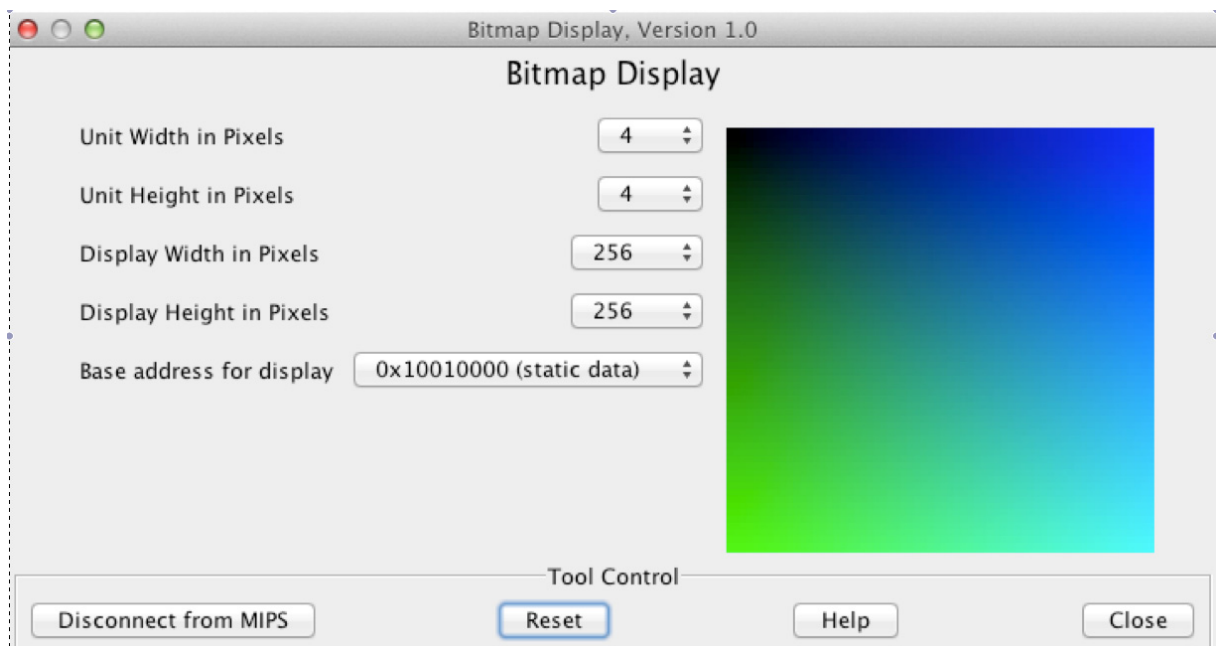


Assim o primeiro pixel (canto superior esquerdo) está no endereço base do display, o segundo pixel da primeira linha 4 bytes à frente, e o primeiro pixel da segunda linha está 64x4 bytes em relação ao endereço base.

- a) Faça um pequeno programa que desenhe um quadrado vermelho no canto superior esquerdo do ecrã de dimensões 20x20 pixels (do simulador). O resultado deverá ser o da figura seguinte:



- b) Escreva um programa que atribua a cada um dos 64x64 pixéis uma cor que seja calculada da seguinte forma: vermelho=0, verde=número da linha multiplicada por 4 e azul=número da coluna multiplicado por 4. O resultado deverá ser o seguinte:



- c) Escreva um programa que desenhe um quadrado vermelho sobre um fundo azul. O quadrado deverá ter dimensões 11x11 e o seu canto superior esquerdo tem coordenadas (10,10). O resultado deverá ser o seguinte:

