EA871 - Relatório 3

Mônica Aoki Faria RA: 156787

Leonardo Rodrigues Marques RA: 178610

1. As instruções THUMB possuem a mesma finalidade das instruções ARM normais com a diferença de que elas são codificadas com 16 bits e não 32. Essas instruções geram códigos menores mas podem aumentar o número de instruções que o processador deve executar. O uso das instruções THUMB juntamente com ARM tem o objetivo de diminuir o tamanho do código e manter o consumo de energia baixo.
2. A diretiva .data diz ao assembler que a próxima sessão é a sessão de dados aonde há declarações de label, tipo e valores de dados.
3. A diretiva .text diz ao assembler que a próxima sessão é a sessão de instruções em linguagem assembly.
4. A diretiva .word é usada para alocar e inicializar espaço para uma variável. O número que procede o .word é o valor inicial da variável.
5. Outro Arquivo
6. push {lr}

Código em binário: 0x00000800: 1011 0101 0000 0000

bits 15-9: código da instrução 1011010 (visto na pg 167)

bit 8: M

bits 0-7: register list

O que a instrução faz: empilha a pilha do sistema

cmp r0,#0

Código em binário: 0x00000802: 0010 1000 0000 0000

bits 15-11: código da instrução 00101 (visto pg 127)

bits 10-8: número relacionado ao registrador r0

bits 7-0: valor referência a ser comparado

O que a instrução faz: compara r0 com zero

beq return

Código em binário: 0x00000804: 1101 0000 0000 0001

bits: 15 - 12: código da instrução: 1101(visto pg 119)

bits: 11 - 8: condição equal

bits 7 - 0: immediate value de return

O que a instrução faz: se os valores comparados em cmp forem iguais ele faz um jump para return

sub r0,#1

Código em binário: 0x00000806: 0011 1000 0000 0001

bits 15-11: código da instrução 00111 (visto pg 185)

bits 10-8: numero relacionado ao registrador r0

bits 7-0: valor imediato a ser subtraído

O que a instrução faz: subtrai o valor de r0 e recoloca-o em r0

b again

Código em binário: 0x00000808: 1110 0111 1111 1101

bits 15-11: código da instrução 11100 (visto pg 119)

bits 10-0: immediate value de again

O que a instrução faz: ela salta para a instrução requerida

pop {pc}

Código em binário: 0x0000080a: 1011 1101 0000 0000

bits 15-9: código da instrução 1011110 (visto pg 165)

bits 8: P

bits 7-0: register list

O que a instrução faz: desempilha a pilha do sistema

1. A diretiva .align 2 indica ao montador para colocar as instruções em número par. Ao retirar a diretiva, o programa apresentará erro pois terá número ímpar de instruções. Isso pode ser corrigido adicionando um instrução adicional como por exemplo ‘nop’. Dessa forma, o programa voltará a ter um número par de instruções. Ao recolocar a diretiva .align 2, o programa continua a funcionar. A diretiva garante a montagem correta do programa.
2. ldr r3,SIM\_SCGC5

Código em binário: 0x0000080c: 0100 1011 0001 0000

bits 15-11: código da instrução 01001 (visto na pg 141)

bits 10-8: número relacionado ao registrador 011

bits 7-10: deslocamento. Offset adicionado ao valor de Align(PC, 4) da instrução para formar o endereço.

O que a instrução faz: Carrega o conteúdo da memória SIM\_SCGC5 em r3.

1. ldr r2,[r3]

Código em binário: 0x0000080e: 0110 1000 0001 1010

bits 15-11: código da instrução 01101 (pg 139)

bits 10-6: valor do deslocamento incrementado ao valor do registrador r3

bits 5-3: número relacionado ao registrador r3

bits 2-0: número relacionado ao registrador r2

O que a instrução faz: Carrega uma palavra encontrada no endereço de memória dado por r3

1. bl delay

Código em binário: 0x00000848: 1111 0111 1111 1111

bits 15-11: código da instrução 11110 (pg 123)

bits 10:S

bits 9-0: immediate value de delay

O que a instrução faz: a instrução faz um salto para outra instrução com link. Ao acioná-la, o curso do programa é desviado para a instrução requerida e depois retorna para o curso original.