1.7

1.7.1 Warm-up questions

1. What is the difference between an ARM processor mode and an ARM processor state?

Processor mode define o conjunto de registradores utilizados e controla acesso a recursos protegidos. Processor state serve para alternar entre o uso de instruções alinhadas a palavras de 32 bits e uso de instruções Thumb alinhadas a halfwords de 16 bits

2. Name the different modes and states of the ARM processor.

Processor modes:

- User
- Fast Interrupt
- Interrupt
- Supervisor
- Abort
- Undefined
- System

3. What register is used for the PC? The LR?

PC: r15, LR: r14

4. What is the normal usage of r13?

Stack Pointer.

5. Which bit of the CPSR defines the state?

Bit T, 6.o bit, bit número 5.

6. What is the difference between the boundary alignments of ARM vs Thumb instructions?

ARM: alinhadas a palavras de 32 bits. Thumb: alinhadas a halfwords de 16 bits.

7. Explain how to disable IRQ and FIQ interrupts.

Utilizando os interrupt disable bits do CPSR I (r7) e F (r6), é possível desabilitar interrupções normais e do tipo fast, respectivamente.

1.7.2 Endianness

Suppose that r0 = 0x12345678 and that this value is stored to memory with the instruction 'store r0 to memory location 0x4000.' What value would r2 hold after the instruction 'load a byte from memory location 0x4000 into r2' when memory is

organized as big-endian? What would r2 hold when memory is organized as little-endian?

- Big-endian:

```
r0 = 0x12345678
store r0 to 0x4000
```

Endereço	Valor
0x4000	0x78
0x4001	0x56
0x4002	0x34
0x4003	0x12

```
load a byte from memory location 0x4000 into r2
r2: 0x78
```

- Little-endian:

```
r0 = 0x12345678
store r0 to 0x4000
```

Endereço	Valor
0x4000	0x12
0x4001	0x34
0x4002	0x56
0x4003	0x78

```
load a byte from memory location 0x4000 into r2
r2: 0x78
```