Tutorial 1

Question 1: (8)
a) In what ways is a microcontroller similar to a computer? (2)
b) What is the purpose of the ST-Link/V2? (2)
c) When the microcontroller boots up, what key step does it have to take before it can start executing your instructions? (2)
d) In general, how much is the program counter incremented by on each clock cycle? Why this amount? (2)

Question 2: (4)
a) Compare and contrast the output files of the assembler and the linker (2)
b) What is the purpose of OpenOCD (2)

Question 3: (3*3 = 9)
For each of the following machine code instructions, what action will the CPU take?
You must give the assembly instruction as well as explain what the instruction does.
Also mention how they affect the APSR.

a)	`000 11 0 0 110 111 101`	(hint: A6.7.3)
b)	`010001 10 0 1111 000`	(hint: A6.7.40)
c)	`000 00 00000 000 000`	(hint: A6.7.40)

Question 4: (4)
a) If we want to place the number 1234 effective address 0x2000 001A, which address takes on which byte? (2)
b) If our program consists of 42 instructions, and the first instruction was placed at memory address 0x0800 0042, what would the first free address after the block of instructions be? (1)
c) Would it be possible to start our block of instructions at address 0x0800 0045? Why or why not? (marks for reason) (1)

Question 5: (4)
a) Why is it necessary to have an unconditional branch at the end of our programs? (2)
b) Would the following branch be taken or not? Explain. Marks for explanation. (2)

```
 SUBS  R1,  R4,  R4
 BEQ   foobar
```

Bonus: (2)
We've hidden some complexity when explaining how PC-relative offsets work.
Assume we have the instruction *LDR R0, foo* located at address 0x0800 0042 and the data which has been labeled as *foo* present at effective address 0x0800 0080.
What is the 8-bit immediate data which is encoded as the offset operand in the machine code instruction?

Marked out of: 29 Available marks: 31