Sean Penney and Paul Atkinson
April 18, 2014
CS 472 HW 3

(3.1) The program counter is not a counter. The PC holds, or points to, the memory address of the next instruction to be executed.

(3.2) a. PC points to the memory address of the next instruction to be executed.

b. MAR stores the address that is being accessed by read/write operations.

c. MBR is the memory buffer register, which holds data that has been read from main memory or will be written to main memory.

d. The IR (instruction register) holds the instruction currently being executed.

(3.3) a. $C = 0$, $Z = 0$, $V = 0$, $N = 0$

b. $C = 1$, $Z = 1$, $V = 0$, $N = 0$

c. $C = 0$, $Z = 0$, $V = 0$, $N = 0$

d. $C = 1$, $Z = 0$, $V = 0$, $N = 0$

e. $C = 0$, $Z = 0$, $V = 0$, $N = 1$

f. $C = 1$, $Z = 0$, $V = 0$, $N = 1$

(3.10)

(3.17) Using an 8-bit format for the integer and a 4-bit alignment field allows for a larger range of values. The disadvantage to this mechanism is that there are gaps in the range of values.

(3.18)

(3.19) Using the XOR swap algorithm:

```
EOR r0, r0, r1
EOR r1, r1, r0
EOR r0, r0, r1
```

(3.25)

(3.39)

```
LOOP LDRB r2, [r0], #1 ; get address of next character
STRB, r2, [r1], #1 ; store contents at r2 in r1
TEQ r2, #0 ; check if at end of string
BRNE LOOP ; if not at end of string, go back to loop
```

(3.51)