Alan Palayil

ECE 242 Exercise #6

1.

i.
```
CMPA.L $504,A2
BLT.L Less
BGT.L More
```

ii.
```
CMPA.L $504,A2
BEQ.L Equal
```

```
ORG     $2000
START:
    MOVE.L #20,$504       //Adding the value in memory address $504
    CMPA.L $504,A2        //Comparing the values in $504 and A2
    BEQ.L Equal           //Branch if the values are equal
    BLT.L Less            //Branch if the value in $504 is larger
    BGT.L More            //Branch if the value in A2 is larger
Equal:
    MOVEA.L A2,A1         //Move the content of A2 to A1
    ADDA.L $504,A1        //Add the contents of $504 to A1
    SIMHALT               //Halt simulator
More:
    MOVE.L A2,D0          //Move the content of A2 to D0
    MOVE.L $504,D1        //Move the content of $504 to D1
    SIMHALT               //Halt simulator
Less:
    MOVE.L $504,D0        //Move the content of $504 to D0
    MOVE.L A2,D1          //Move the content of A2 to D1
    SIMHALT               //Halt simulator
 END    START         ; last line of source
```

2.

i. $1011_2 = (1 * 2^3) + (0 * 2^2) + (1 * 2^1) + (1 * 2^0)$

$= 8 + 0 + 2 + 1$

= 11

$0.101_2 = \left(1 * \frac{1}{2}\right) + \left(0 * \frac{1}{2^2}\right) + \left(1 * \frac{1}{2^3}\right)$

= 0.625

Thus, 1011.101_2 = 11.625

ii. 747_{10} divided by 16

747/16 with remainder 11 (B)

46/16 with remainder 14 (E)

2/16 with remainder 2 (2)

Thus, the $747_{10} = 2EB_{16}$

iii. $6C_{16}$ in 16-bit sign extend is $006C_{16}$ or $0000\ 0000\ 0110\ 1100_2$

iv. -1809_{10} in 16-bit binary $=(1000011100010001)_2$

So, 1’s complement of -1809_{10} = $(1111100011101110)_2$

So, 2’s compliment of -1809_{10} = $(1111100011101111)_2$

v. $64.25_{10} = 1000000.01_2$

1000000.01= 1. 00000001*2^6

Sign (S) = 1

(1.N)*2^{E-1023}

Mantissa (N)=00000001

E-1023=6

E=1029

1029_{10}= 10000000101_2

IEEE Standard floating point double precision representation is:

S	E	N
1	10000000101	000000010000000000000000…
1 Bit	11 Bit	52 Bit

3.

i. D1= $0000 FF02, Condition code: X=1, N=0, Z=0, V=0, C=1

ii. D1= $0001 0001, Condition code: X=0, N=0, Z=0, V=0, C=0

iii. D1= $0001 FFFE, Condition code: X=0,1 N=1, Z=0, V=0, C=0

iv. D1= $0000 0001, Condition code: X=1, N=0, Z=0, V=0, C=1

v. D1= $0000 0000, Condition code: X=0, N=0, Z=1, V=0, C=0

4. Using logical shift of $1110\ 0101_2$ = 00011100_2

$= 1 * 2^4 + 1 * 2^3 + 1 * 2^2$

= 28

5.

i. A number will be odd if the last or least significant bit in a binary number is set.

BTST.L #$00, D0

ii. A signed number has the MSB as 1 and as m bits long is signed.

BTST.L #$m, D0

iii. We test 5th bit; we can identify the upper case characters.

BTST.L #$05, D0