Tutorials

ECE 511: Digital System & Microprocessor
Tutorial 3
DRD

- Transfer data $AB4432F5 to D0.
- Transfer 1 byte from D0 to D1. What value is copied to D1?
- Transfer 1 word from D0 to D2. What value is copied to D1?
- Copy all data in D0 into D7 (long). D7 value should be the same as D0.
ARD

- Move long value $400400 to address register A5 using:
  - MOVEA (Move Address)
  - LEA (Load Effective Address)

- Move word value $A123 to A0 using:
  - MOVEA (Move Address)
  - LEA

- Load D0 = $5465A766, then transfer word data in D0 to A1.
ARI + PD

- Execute these instructions step-by-step and check A7 and the stack output:
  - MOVE.L #$11112222,-(A7)
  - MOVE.L #$33334444,-(A7)
  - MOVE.L #$55556666,-(A7)
  - MOVE.L #$77778888,-(A7)

- (Stack contents can be viewed using View>Stack in Sim68k window)
Use ARI + PI to input these data into memory starting from address $4000.

- $45
- $6A
- $341A
- $ACBA1234

Be sure to check the address register value after each instruction executes.
Send binary value 01101000 to 1 byte in D0. What hex value is stored in D0?

Send octal value 45 to 1 word in D3. What hex value is stored in D3?

Send hexadecimal value $45678901 to address $4500. What hex value is stored in $4500?
ID

Send decimal value 222,331 to memory location $400400. What hex value is stored in $400400?

Send characters ‘BUKU’ to address $4000, ‘SAYA’ to address $4010, ‘BUKU’ to address $4020, and ‘BARU’ to address $4030. Check the output in memory.
ARI + D

Send characters ‘BUKU’ to address $4000, ‘SAYA’ to address $4010, ‘BUKU’ to address $4020, and ‘BARU’ to address $4030 using ARI + D. Check the output in memory.
Use ARI + 1 to store the following information inside memory locations starting from $3000.

<table>
<thead>
<tr>
<th>Name</th>
<th>Money</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALI</td>
<td>RM50</td>
</tr>
<tr>
<td>ABU</td>
<td>RM100</td>
</tr>
<tr>
<td>ATAN</td>
<td>RM 75</td>
</tr>
<tr>
<td>ANEP</td>
<td>RM 20</td>
</tr>
</tbody>
</table>
Use IA to set the value of the Status Register so that:

- Trace bit is on.
- Interrupt Mask Bits is set to level 5.
- All CCR values are cleared.
Tutorial 4
Question 1

- Write a program that stores 15,000 inside memory address $5000, and 3,400 inside memory address $4300.

- Then, put both values from memory into D0 and D1.

- Add together the values stored inside D0 and D1, and store the value inside D1. Examine the value in D1 and CCR.
Question 2

- Store values $25, $41, $66, $34, and $\text{A}2$ in memory address starting from $4000$.
- Add together the values and store inside D3.
- What is the value stored in D3?
Question 3

- Put the value of $35 inside D5.
  - Subtract $30 from D5. What is the result and CCR value?
  - Then, subtract $10 from D5. What is the result and CCR value?
Question 4

- Write a program that calculates the area of a circle. The radius is 10 and is stored inside D0. Use $\pi = \frac{22}{7}$. 
Question 5

- Write a program that implements a function similar to the NEG (negate) function.
- Test your answer by comparing it with the NEG function. Are the results the same?
Question 6

- Write a program that tests whether the byte value inside D7 is positive or negative.
Question 7

- Put value $FFFFAAAAA$ inside D1.
- Write a program that clears the upper word in D1 so that the new value inside D1 is $0000AAAA$. 
Question 8

- Put the value $FFFF0000 into D3.
- Use logical shift so that the new value in D3 is $000000FF.
- Examine and explain the CCR value after the instruction executes.
Question 9

- Input the value $00FF00FF$ into D0
- Use logical operations (AND, OR, NOT, XOR) so that the value in D0 becomes $FF00FF00$. 
Question 10

- Write a program that determines your age when you input your date of birth in D5.
Question 11

Write a program that adds BCD numbers 65 and 76 together. Since BCD operations also involve the X bit, you must clear the X bit first before you add the numbers together.
Question 12

- Write a program that finds the 9’s complement value of BCD value 58.
Question 13

- Write the instruction needed to add the lower word of register D0, with the upper word in D6. The result saved in D6.
Question 14

- Multiply the contents of D3 by 0.125. Since fractional multiplication is not available you must think of an alternative way to solve this problem. Assume that D3 contains an unsigned binary value.
Question 15

- Write a subroutine that calculates the volume of a cube whose length on one side has been placed in register D3. The volume should be in D2 when finished.
Question 16

- Write the instructions to compute the area of a right triangle whose side lengths are stored in D2 and D3. Load the result in D1.
Question 17

- Write a program that produces 2, 4, 6, …., 20 inside memory locations starting from address $5000.
Question 18

Modify the original program in Q17 to calculate the sum of \(2 + 4 + 6 + \ldots + 20\).
Tutorial 5
Question 1

- Write a program that converts x inches to cm. The inches value is stored inside D1.
- Hint: 1 inch = 2.54 cm.
Question 2

- Write a program that checks the long-word value in D3.
  - If the long-word value is negative, then find its 2’s complement value.
  - If the long-word value is positive, then clear the value inside D3.
Question 3

- Write a program that tests the byte value in memory location $4002.$
  - If the byte value is within the range of $3 \leq x \leq 10,$ then don’t change its value.
  - If the value is more than 10, then set its value to 10.
  - If the value is less than three, then set its value to 3.
Question 4

- Write a program that writes the character ‘A’ to memory locations $2000 to 200F, ‘B’ to memory locations $2010 to $201F, and ‘U’ to memory locations $2020 to $202F.

- All characters should be entered using for loops.
Question 5

- Write a program that reads a number from D0, and writes the number from 1 to D0 in memory locations starting from $5000.
Question 6

- Write a program that stores even numbers in the range from 1 to 100 in memory locations starting from $3000.$
Question 7

- Next, write a program that automatically converts the character values in the memory locations into its hexadecimal value equivalent.