**Assignment 1**

#include "msp430.h"

#include "stdio.h"

int main(void)

{

WDTCTL = WDTPW + WDTHOLD; //stop the watchdog timer

int v = 12; //the base

int p = 4; //the exponent, must be positive and 1 or greater

int answer = 1; //the solution

int count = 1; // initializing the counter

//Loop begins

while(count <= p)

{

answer = answer\*v; // calculate the answer

count++; // increment the counter

}

printf("%d raised to the %d power is = %d", v, p, answer); // print message

return answer; // return the result

}

It takes the demo code with recursion 457112 clock cycles to complete while it takes assignment 1 8065 clock cycles. The difference is due to the steps inside those function calls in the recursion which recursion we don’t have in the loop of assignment 1.

Declare base value

Declare power value

For loop

Answer = answer\*value

Increment the counter

Print answer message after the loop is exited

Exit program

Assignment 1 flow chart

**Assignment 2**

#include "msp430.h"

#include "stdio.h"

int main(void)

{

WDTCTL = WDTPW + WDTHOLD; //stop the watchdog timer

for(i = 1; i <= 5; i++) // initializing the for loop

{

printf("\nSample message, Counter = %d", i); // print message each time the counter change

if(i == 2)

{

printf("\nExtra message, Counter = %d", i); // print extra message if counter eqauls 2

}

}

return 0;

}

Loop initializer

Print message

Increment the counter

Print an extra message if counter = 2

If counter different than 2

Exit the program if counter > 5

Flow chart for Assignment 2