**Task 1: A comprehensive commented file of the original code**

#include <stdint.h>

#include <stdbool.h>

#include <math.h>

#include "inc/hw\_memmap.h"

#include "inc/hw\_types.h"

#include "driverlib/fpu.h"

#include "driverlib/sysctl.h"

#define TARGET\_IS\_BLIZZARD\_RB1

#include "driverlib/rom.h"

// Define PI

#ifndef M\_PI

#define M\_PI 3.14159265358979323846

#endif

#define SERIES\_LENGTH 100 // data buffer depth

float gSeriesData[SERIES\_LENGTH]; // Array of floats SERIES\_LENGTH

int32\_t i32DataCount = 0; // loop counter

int main(void)

{

float fRadians; // Variable to calculate sine

ROM\_FPULazyStackingEnable(); // Turn on Lazy Stacking

ROM\_FPUEnable(); // Turn on FPU

// Set System Clock to 50MHz

ROM\_SysCtlClockSet(SYSCTL\_SYSDIV\_4 | SYSCTL\_USE\_PLL | SYSCTL\_XTAL\_16MHZ | SYSCTL\_OSC\_MAIN);

// Divide 2\*pi by array length since a full sine wave cycle is 2\*pi radians

fRadians = ((2 \* M\_PI) / SERIES\_LENGTH);

// 100 sine values are calculated

while(i32DataCount < SERIES\_LENGTH)

{

// sine value stored in array

gSeriesData[i32DataCount] = sinf(fRadians \* i32DataCount);

// Incrementation next value

i32DataCount++;

}

while(1)

{

}

}

**Task 2: Modified code to implement new equation to generate a frequency of 5 Hz.**

**1.0\*sin(2p50t) + 0.5\*cos(2p 200t)**

**#include** <stdint.h>

**#include** <stdbool.h>

**#include** <math.h>

**#include** "inc/hw\_memmap.h"

**#include** "inc/hw\_types.h"

**#include** "driverlib/fpu.h"

**#include** "driverlib/sysctl.h"

**#define** TARGET\_IS\_BLIZZARD\_RB1

**#include** "driverlib/rom.h"

// Define PI

**#ifndef** M\_PI

**#define** M\_PI 3.14159265358979323846

**#endif**

**#define** SERIES\_LENGTH 1000 // Depth of the data buffer

**float** gSeriesData[SERIES\_LENGTH]; // Array of floats SERIES\_LENGTH long

int32\_t i32DataCount = 0; // Counter for the loop

**int** **main**(**void**)

{

**float** fRadians; // Variable to calculate sine

ROM\_FPULazyStackingEnable(); // Turn on Lazy Stacking

ROM\_FPUEnable(); // Turn on FPU

// Set System Clock to 50MHz

ROM\_SysCtlClockSet(SYSCTL\_SYSDIV\_4 | SYSCTL\_USE\_PLL | SYSCTL\_XTAL\_16MHZ | SYSCTL\_OSC\_MAIN);

// Divide 2\*pi by array length since a full sine wave cycle is 2\*pi radians

fRadians = ((2 \* M\_PI) / SERIES\_LENGTH);

// Calculate 1000 formula values

**while**(i32DataCount < SERIES\_LENGTH)

{

// Store formula values into array

gSeriesData[i32DataCount] = **sinf**(fRadians \* 50 \* i32DataCount) + 0.5\***cosf**(fRadians \* 200 \* i32DataCount);

// Increment to go to next value calculation

i32DataCount++;

}

**while**(1)

{

}

}