**Date Submitted: 11/12/19**

**Task 00: Execute provided code**

#define TARGET\_IS\_BLIZZARD\_RB1

#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"

#include "driverlib/rom.h"

#include "driverlib/rom\_map.h"

#ifndef M\_PI

#define M\_PI 3.14159265358979323846

#endif

#define SERIES\_LENGTH 100

float gSeriesData[SERIES\_LENGTH];

int32\_t i32DataCount = 0;

int main(void)

{

float fRadians;

ROM\_FPULazyStackingEnable();

ROM\_FPUEnable();

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

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

while(i32DataCount < SERIES\_LENGTH) // number of data points is under series length

{

gSeriesData[i32DataCount] = sinf(fRadians \* i32DataCount); assign value to array

i32DataCount++;

}

while(1)

{

}

}

**------------------------------------------------------------------------------------**

**Task 01:**

**Modified Code:**

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

**#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"**

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

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

**#ifndef M\_PI**

**#define M\_PI 3.14159265358979323846**

**#endif**

**#define SERIES\_LENGTH 100**

**float gSeriesData[SERIES\_LENGTH];**

**int32\_t i32DataCount = 0;**

**int main(void)**

**{**

**float fRadians;**

**ROM\_FPULazyStackingEnable();**

**ROM\_FPUEnable();**

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

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

**while(i32DataCount < SERIES\_LENGTH) // number of data points is under series length**

**{**

**gSeriesData[i32DataCount] = sinf(fRadians \* i32DataCount); assign value to array**

**i32DataCount++;**

**}**

**while(1)**

**{**

**}**

**}**

**------------------------------------------------------------------------------------**