

**Date Submitted:** 10/20/19**Task 01:**

Youtube Link: None

Modified Schematic (if applicable): N/A

**Modified Code:**

```

#include <stdint.h>
#include <stdbool.h>
#include <stdlib.h>
#include <math.h>
#include "inc/hw_memmap.h"
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/pin_map.h"
#include "driverlib/fpu.h"
#include "driverlib/sysctl.h"
#include "driverlib/debug.h"
#include "driverlib/rom.h"

#ifndef M_PI
#define M_PI 3.14159265358979323846
#endif

#define SERIES_LENGTH 100

float gSeriesData[SERIES_LENGTH]; //Define array for sine values

int32_t i32DataCount = 0; //Counter for sine wave

int main(void)
{
    float fRadians;
    //Enable Lazy Stack
    ROM_FPULazyStackingEnable();
    //Turn on FPU
    ROM_FPUEnable();

    //Set Clock to 50MHz
    ROM_SysCtlClockSet(SYSCTL_SYSDIV_4 | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ |
SYSCTL_OSC_MAIN);
    //(2pi rads)/ 100 to get full cycle of sine wave
    fRadians = ((2 * M_PI) / SERIES_LENGTH);
    //loop until sine wave is calculated
    while(i32DataCount < SERIES_LENGTH)
    {
        gSeriesData[i32DataCount] = sinf(fRadians * i32DataCount); //calculate sine
wave values
        i32DataCount++; //get next data value
    }
}

```

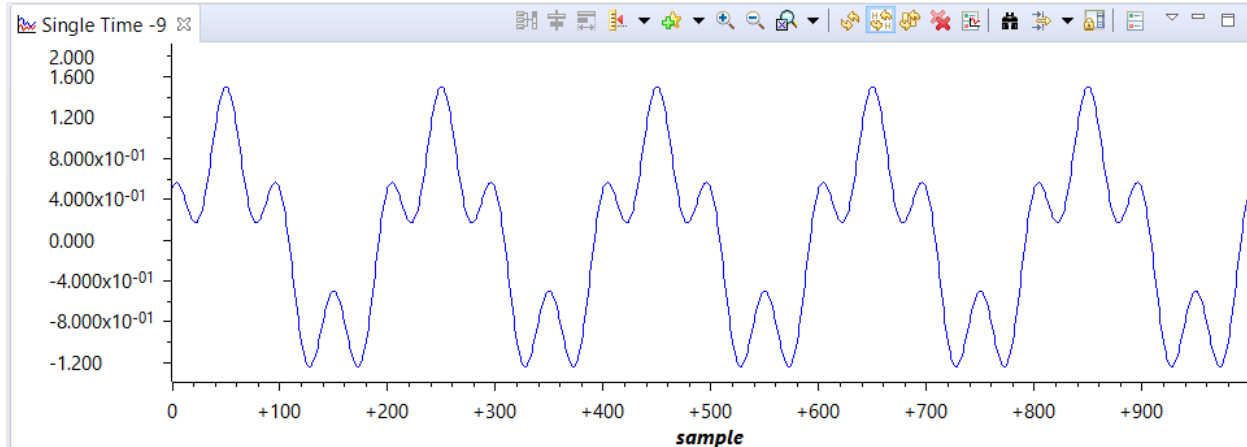
**Grading scheme:** 30% Coding, 30% Documentation, 40% Execution/Video.

```
}  
  
while(1)  
{  
    //loop continuously  
}  
}
```

---

**Task 02:**

Youtube Link: None



Modified Schematic (if applicable): N/A

**Modified Code:**

```
#include <stdint.h>
#include <stdbool.h>
#include <stdlib.h>
#include <math.h>
#include "inc/hw_memmap.h"
#include "inc/hw_types.h"
#include "driverlib/gpio.h"
#include "driverlib/pin_map.h"
#include "driverlib/fpu.h"
#include "driverlib/sysctl.h"
#include "driverlib/debug.h"
#include "driverlib/rom.h"

#ifndef M_PI
#define M_PI 3.14159265358979323846
#endif

#define SERIES_LENGTH 200

float gSeriesData[SERIES_LENGTH*5]; //Define array for sine values

int32_t i32DataCount = 0; //Counter for sine wave
int main(void)
{
    float fRadians;
    //Enable Lazy Stack
    ROM_FPULazyStackingEnable();
    //Turn on FPU
    ROM_FPUEnable();

    //Set Clock to 50MHz
    ROM_SysCtlClockSet(SYSCTL_SYSDIV_4 | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ |
SYSCTL_OSC_MAIN);
```

**Grading scheme:** 30% Coding, 30% Documentation, 40% Execution/Video.

```
fRadians = ((2 * M_PI) / SERIES_LENGTH);  
//loop until sine wave is calculated  
  
while(i32DataCount < SERIES_LENGTH*5)  
{  
    gSeriesData[i32DataCount] = sinf(fRadians * i32DataCount) + .5 *  
    cosf(fRadians * 4 * i32DataCount); //calculate values  
    i32DataCount++;  
}  
  
while(1)  
{  
    //loop continuously  
}  
}
```

---