LAB-6 REPORT

Subject: **Embedded Hardware Design**

Subject Code: **EL203**

Members: 1) Shantanu Tyagi **(201801015)**
2) Nikhil Mehta **(201801030)**
3) Sudhanshu Mishra **(201801114)**
4) Bhavana Kolli **(201701082)**

Problem Statement

Approximate the sine function using its Taylor expansion given by the following equation:

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!}$$

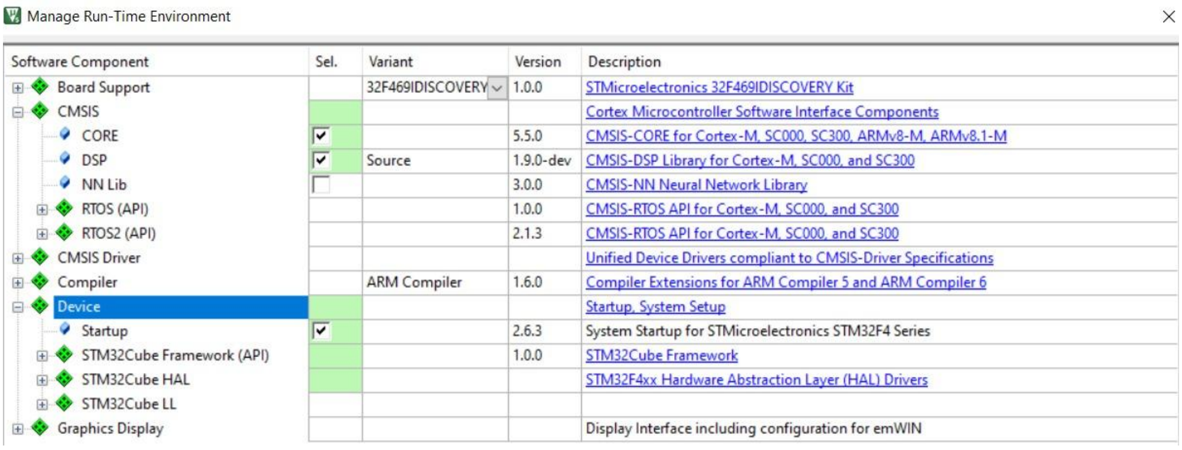
Environment Setup

In the IDE, install the Keil::STM32F4xx DFP Package from Pack Manager.

1) Create a new Project and rename it using the appropriate nomenclature.

   1.1.1.1. Set STM32F407VGTx as the target device under the device header

   1.1.1.2. Modify the options for the target device:



Manage Run-Time Environment

Software Component	Sel.	Variant	Version	Description
Board Support		32F469IDISCOVERY	1.0.0	STMicroelectronics 32F469IDISCOVERY Kit
CMSIS				Cortex Microcontroller Software Interface Components
CORE	☑		5.5.0	CMSIS-CORE for Cortex-M, SC000, SC300, ARMv8-M, ARMv8.1-M
DSP	☑	Source	1.9.0-dev	CMSIS-DSP Library for Cortex-M, SC000, and SC300
NN Lib	☐		3.0.0	CMSIS-NN Neural Network Library
RTOS (API)			1.0.0	CMSIS-RTOS API for Cortex-M, SC000, and SC300
RTOS2 (API)			2.1.3	CMSIS-RTOS API for Cortex-M, SC000, and SC300
CMSIS Driver				Unified Device Drivers compliant to CMSIS-Driver Specifications
Compiler		ARM Compiler	1.6.0	Compiler Extensions for ARM Compiler 5 and ARM Compiler 6
Device				Startup, System Setup
Startup	☑		2.6.3	System Startup for STMicroelectronics STM32F4 Series
STM32Cube Framework (API)			1.0.0	STM32Cube Framework
STM32Cube HAL				STM32F4xx Hardware Abstraction Layer (HAL) Drivers
STM32Cube LL				
Graphics Display				Display Interface including configuration for emWIN

1.2. Create a new group under the project

1.3. Make modifications for the following additional options for the target device:

   1.3.1. Set ARM compiler version 6 under the Target -> Code Generation header

   1.3.2. Select the Simulator radio button and load the KEIL_STM.ini file as an initialization file under the Debug header

1.4. Add the main.c and sinewave.c file under the created group in the project with the relevant header file (#include "stm32f4xx_hal.h",#include "arm_math.h")

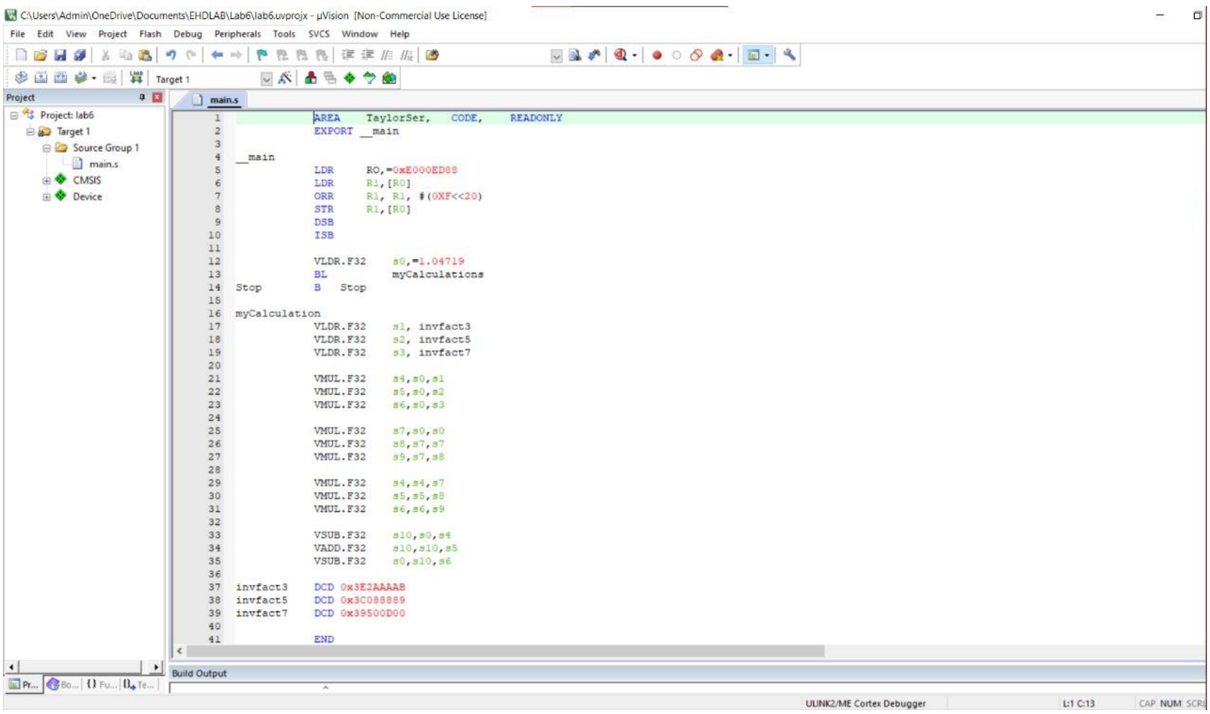
Running the Simulation

1 To run the code, first we have to compile the main.c file. This is done by clicking on the build button in Keil IDE.
2 Once the build is complete, we are prompted with errors and we need to fix them.
3 After fixing the bugs we move to the debug section and run the code to get the results.
4 To see the results, we can either see the result value or check out the graph for the simulation.
5 We can see the values of the variables directly by clicking on the variable and then on ‘add to watch’.
6 In order to get the graph, we select the variable and click on ‘analyze’.
7 After checking the output, we will have to close the debug session to make further changes in the code. Then follow the steps mentioned above again to see the new output.

CODE:

```
//main.s
      AREA TaylorSer, CODE, READONLY
      EXPORT __main

__main
           LDR R0, =0xE000ED88
           LDR R1, [R0]
           ORR R1, R1, #(0xF<<20)
           STR R1, [R0]
           DSB
           ISB

           VLDR.F32   s0, =1.04719
           BL         myCalculation
Stop  B    Stop

myCalculation
                VLDR.F32 s1, invfact3
                VLDR.F32 s2, invfact5
                VLDR.F32 s3, invfact7
```

```
            VMUL.F32 s4, s0, s1;compute x/3!
            VMUL.F32 s5, s0, s2 ;compute x/5!
            VMUL.F32 s6, s0, s3 ;compute x/7!

            VMUL.F32 s7, s0, s0 ;compute x^2
            VMUL.F32 s8, s7, s7 ;compute x^4
            VMUL.F32 s9, s7, s8 ;compute x^6

            VMUL.F32 s4, s4, s7 ;x^3/3!
            VMUL.F32 s5, s5, s8 ;x^5/5!
            VMUL.F32 s6, s6, s9 ;x^7/7!

            VSUB.F32 s10, s0, s4
            VADD.F32 s10, s10,s5
            VSUB.F32 s0, s10, s6


invfact3    DCD   0x3E2AAAAB ; = 1/3!
invfact5    DCD   0x3C088889 ; = 1/5!
invfact7    DCD   0x39500D00 ; = 1/7!

            END
```



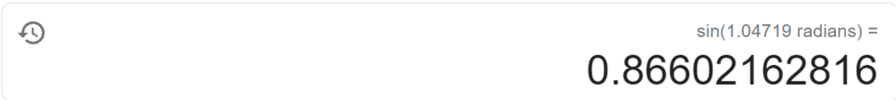
Conclusions/Observations:

1. The value that we get after executing the code is in hexadecimal format.

2. After this value is converted into the corresponding decimal value, it comes out to be equal to 0.86601.

3. This value is the sine of the value we had in the beginning, i.e. 1.04719.

4. The expected value which is the actual value is 0.86602 which is very close to our approximated value using the first 4 terms from the Taylor series of the sine function.



5. This formula works for all x in the range of [0, pi/2].