Introduction

The code for this project was implemented using the STMCubeIDE for which all the dependencies can be found here: https://github.com/NPLSHA003/3088F_Project/tree/main/Microcontroller_interfacing

The files should be downloaded and loaded using STMCubeIDE. The files has all the functions needed to run the scripts and can be found here: https://github.com/NPLSHA003/3088F_Project/blob/main/Microcontroller_interfacing/SRC

The Includes:

#include "stm32f0xx_hal.h"

Contains the libraries used for the communications protocol namely the I2C and UART protocols

#include "string.h"

Used for direct communication with the EEEPROM via strings printed to the display

#include "stdio.h"

Standard input output from the C library to allow to prompt user for inputs and give outputs

The STMCubeIDE generates a template that initializes the functions required for the STM.

init_Sensor(void): gets the sensor started with default parameters

MX_I2C2_Init(void): Sets up I2C connectivity in order to communicate with the sensor.

MX_I2C1_Init(void): Sets up I2C connectivity in order to communicate with the EEPROM.

MX_GPIO_Init(void): sets up the GPIO pins on the STM Board

SystemClock_Config(void): Initialises the RCC Oscillators

MX_USART1_UART_Init(void): Initialises I2C communication for FTDI chip to USB

Communication and readings from EEEPROM:

Two functions written to communicate with the EEEPROM

```
void EEPROM_write(uint8_t Data, uint8_t mem_Address):
```

The function allows user to store information at an EEEPROM address with the standard EEEPROM address being 0x50

Data is the variable to be stored which is 8-bit mem_address is the memory address specified by the user to store data in

```
 void EEPROM_read(uint8_t mem_Address):
```

Collects and reads the data at a specific memory address retrieved it is a printed 8-bit value