ELECTRICAL AND ELECTRONICS
ENGINEERING DEPARTMENT

EE447 Introduction to Microprocessors

Preliminary Report

Audio Frequency Based Stepper Motor Driver

Name	**:** Mustafa Barış Emektar
ID	**:** 2304533
Sec	**:** 1 / TH-2

Introduction

This project is expected to build a system that drives the step motor based on the frequency of audio signals that it catches. Same as the previous experimental works, the TM4C123G Board will be used. The other components that will be used in the project are listed as follows: GY-MAX9814 Microphone Module, NOKIA 5110 LCD Screen, 1 Potentiometer (optional), one 4x4 Keypad (optional), 2 Buttons and RGB LED Placed on the TM4C123G Board, Stepper Motor.

The project consists of some major parts covered in the lectures during the semester. After implementing these parts, assembling these parts is the last and easiest step to do. In this pre-report, the explanation of these parts (as a summary) will be given. Also, the flowchart and the pseudo-codes are going to be shared as well as the estimated pinout of the system. The parts (subunits) of the system are as follows:

- Audio Sampling

  The system will listen to the audio signal in an infinite loop at a constant sampling frequency. The samples are stored in an array of 256 elements. The system calculates the frequency of the samples when the array is filled and updates the output accordingly.

- Frequency Detection

  This sub-unit detects the frequency of samples collected in the audio sampling sub-unit. 256 point FFT should be used for this process. ARM CMSIS DSP library will be the backbone of that part. If the amplitude of dominant frequency signals is under the amplitude threshold, all LEDs must be turned off, and the stepper motor continues its rotation with its current speed.

- Stepper Motor Driving

  If the amplitude is higher than the threshold, the related LED will be turned on according to frequency thresholds, and the stepper motor's speed will be adjusted proportionally to the frequency. Also, the brightness of the LED can be adjusted according to amplitude as a bonus. Moreover, adjustment of the frequency thresholds with user input is also a bonus.

  Moreover, the adjustment of the direction of the motor belongs to this sub-unit. The program will create an interrupt with the press of one of the buttons, SW1 or SW2. Then the direction of the motor will be adjusted accordingly.

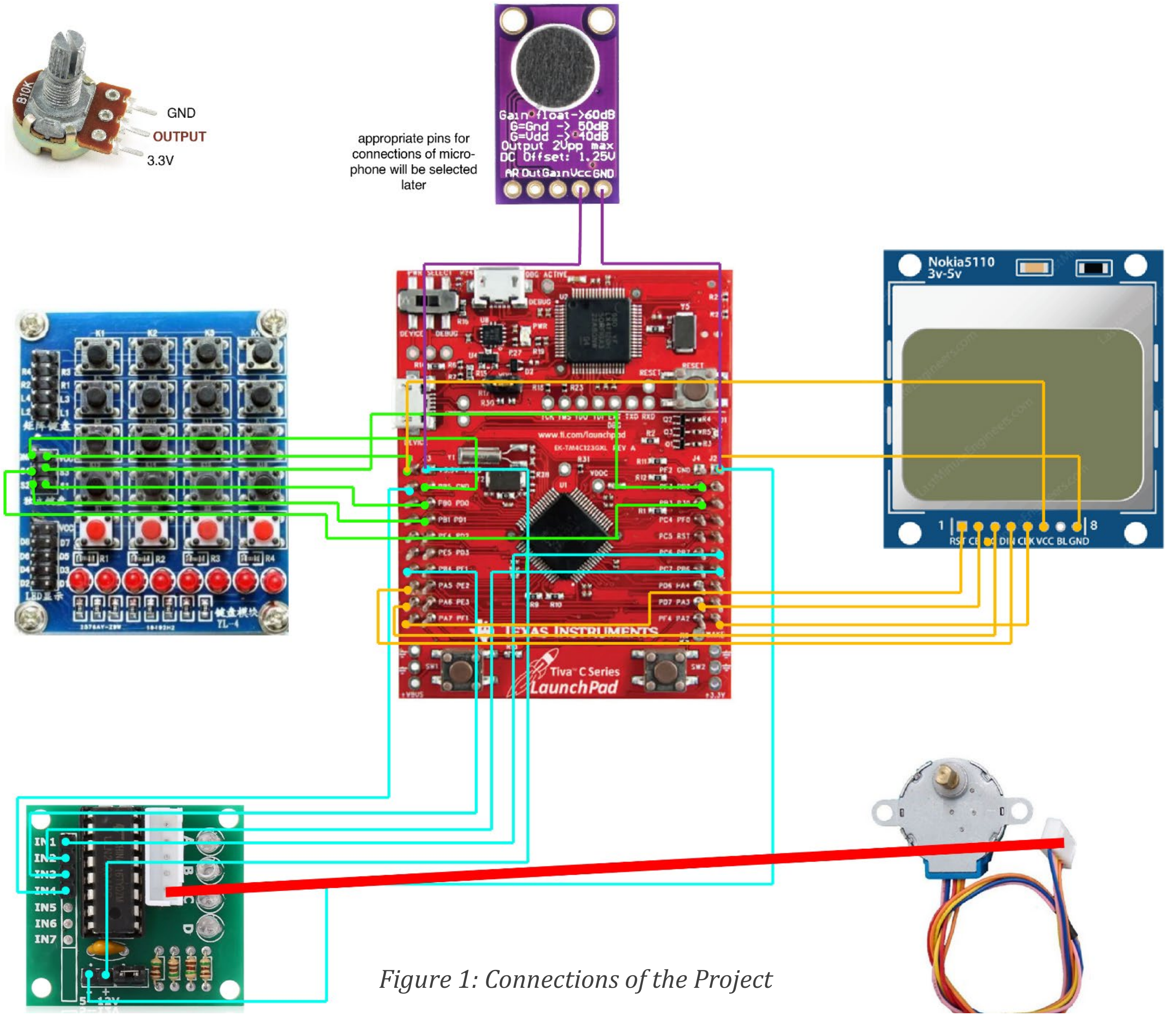
- User Interface

  In this part, the outputs will be shown in the screen and updated continuously. The outputs are as follows: frequency, amplitude and their thresholds.

Overall System

The overall system has divided into four sub-units, as stated above. The functionalities of the sub-units are well defined and separated from each other. The detailed flow chart for the whole project has shown on the following page. The red rectangles indicate the output, and the green rectangles indicate the inputs.

Also, the pinout of the system has been shared on the last page. Connections of most of the parts have been shown. However, connections of some parts like the microphone will be decided later.



appropriate pins for connections of microphone will be selected later

Figure 1: Connections of the Project

```
Initialize one of the ADC ports
Initialize Systick interrupt
WHILE true
        WAIT 500µs (due to 2kHz sampling rate)
        LISTEN ADC port
        WHILE true until 256 samples collected
                STORE the data in array
```

Figure 2: Pseudocodes for ***Audio Sampling Sub-Unit***

```
Detect the frequency of samples using FFT
Detect the amplitude of samples
IF amplitude of the signal is lower than the threshold
        TURN OFF the LEDs
ELSE (if amplitude of the signal is higher than the threshold)
        (Stepper Motor Driving Sub-Unit)
CONVERT frequency into Hz and send it to display
```

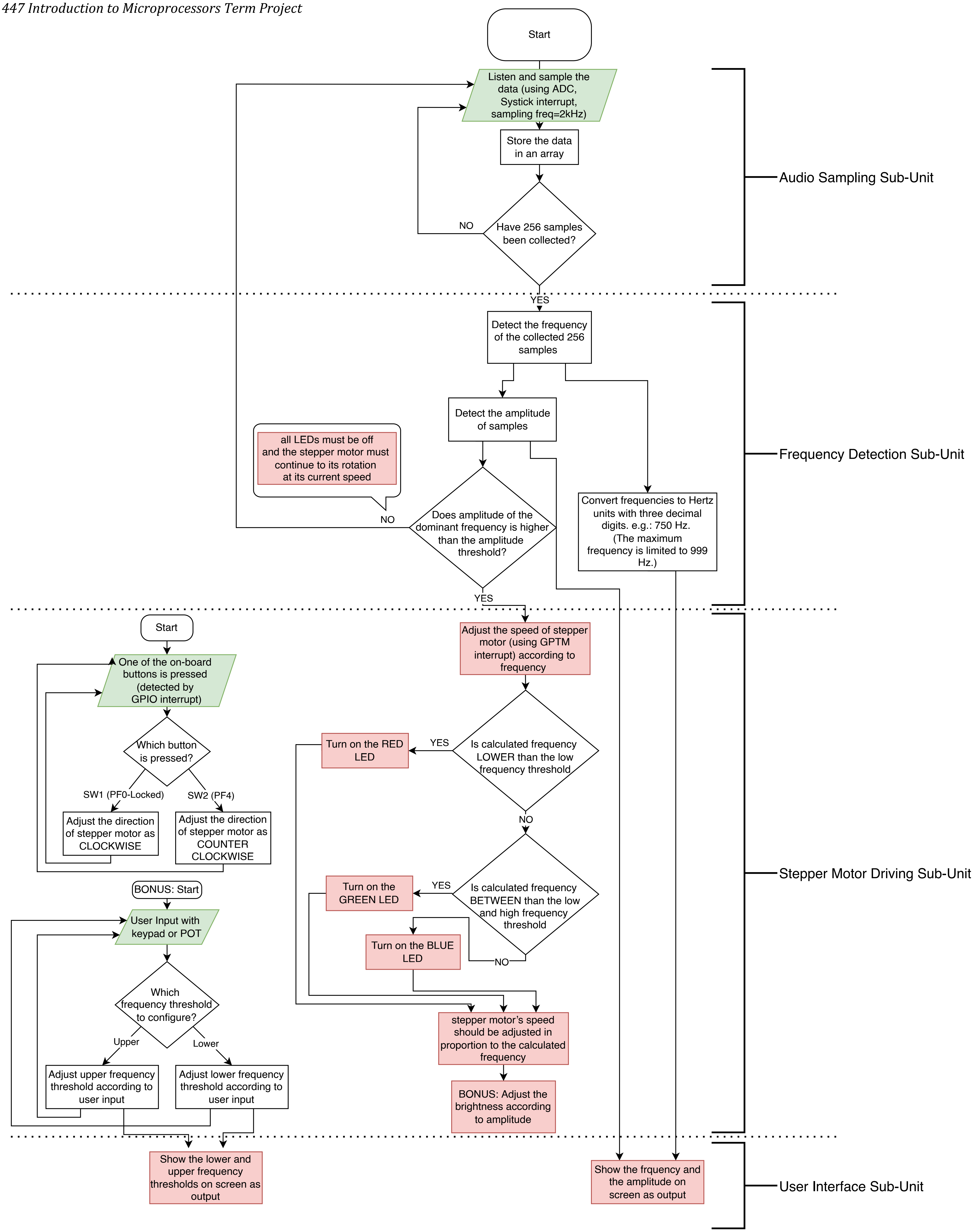
Figure 3: Pseudocodes for ***Frequency Detection Sub-Unit***

```
IF frequency is lower than lower threshold
        TURN ON red led
ELSE IF frequency is between thresholds
        TURN ON green led
ELSE
        TURN ON blue led
Adjust speed of the motor according to frequency
Adjust the brightness according to amplitude(BONUS)
WHILE always
        LISTEN user input (keypad or POT)
        Configure the related threshold
INTERRUPT (GPIO) adjust rotation according to buttons
```

Figure 4: Pseudocodes for ***Stepper Motor Driving Sub-Unit***

```
WHILE true
        Update the frequency
        Update the amplitude
        Update the frequency thresholds
        Update the amplitude thresholds
```

Figure 5: Pseudocodes for ***User Interface Sub-Unit***


Start
Listen and sample the data (using ADC, Systick interrupt, sampling freq=2kHz)
Store the data in an array
Have 256 samples been collected?
NO
YES
Audio Sampling Sub-Unit
Detect the frequency of the collected 256 samples
Detect the amplitude of samples
all LEDs must be off and the stepper motor must continue to its rotation at its current speed
NO
Does amplitude of the dominant frequency is higher than the amplitude threshold?
YES
Convert frequencies to Hertz units with three decimal digits. e.g.: 750 Hz. (The maximum frequency is limited to 999 Hz.)
Frequency Detection Sub-Unit
Start
One of the on-board buttons is pressed (detected by GPIO interrupt)
Which button is pressed?
SW1 (PF0-Locked)
SW2 (PF4)
Adjust the direction of stepper motor as CLOCKWISE
Adjust the direction of stepper motor as COUNTER CLOCKWISE
BONUS: Start
User Input with keypad or POT
Which frequency threshold to configure?
Upper
Lower
Adjust upper frequency threshold according to user input
Adjust lower frequency threshold according to user input
Adjust the speed of stepper motor (using GPTM interrupt) according to frequency
Is calculated frequency LOWER than the low frequency threshold
YES
Turn on the RED LED
NO
Is calculated frequency BETWEEN than the low and high frequency threshold
YES
Turn on the GREEN LED
Turn on the BLUE LED
NO
stepper motor's speed should be adjusted in proportion to the calculated frequency
BONUS: Adjust the brightness according to amplitude
Stepper Motor Driving Sub-Unit
Show the lower and upper frequency thresholds on screen as output
Show the frquency and the amplitude on screen as output
User Interface Sub-Unit