StepMate : PiezoElectric chip based step counter

Amartya Basu, Vinit Yadav, Vishesh Kumar, Harsha
*Course 6650 : Indian Institute of Technology Madras (IITM), Chennai

***Abstract*—Step counting is used for different types of activity monitoring like walking, running etc. Most of the applications relies upon a external wearable sensor that can provide results. However these devices are expensive, energy consuming and sometimes can provide inaccurate results. Carrying such kind of devices is also a problem. In this course project we present StepMate a piezoelectric based step counter which can track human activity. Unlike other devices StepMate doesn't need to be externally carried. It can be easily placed inside a shoe sole and can provide data for monitoring activities.**

***Index Terms*—Piezoelectric chips, micro controller, interfacing.**

I. Introduction

With the advancement in connected systems, there is an increase in numerous activity monitoring applications. Activity monitoring applications can be used to detect physical activities like walking, running etc and can also be used for home care like fall detection, intrusion detection etc. Physical monitoring systems normally relies upon an external wearable device that produces different data based on the activity being done. These kind of devices are very expensive and may not produce accurate results every time. Also these devices need to be carried with the user so that it can perform monitoring. In activities like sports, wearing such kind of external devices is not always feasible. Hence there is a need for a monitoring system which is cheap and can be carried easily. In this project we present StepMate a piezoelectric based step counter that can accurately count the number of steps and monitor the user activity. StepMate can be placed inside a shoe sole and hence doesn't need to be carried externally. In the following sections we have explained about the sensing modalities of StepMate and its overall architecture. At the end we have shown some actual results of StepMate and its future scope. As per our knowledge StepMate is a first of its kind that can be used to classify activities by using piezoelectric chips.

II. StepMate in a nutshell

StepMate utilizes piezoelectric chips along with a micro controller to monitor the activities. We have build an android app that can be used to take the data from these devices and classify the user's physical activity. Overall, StepMate can be broadly classified into 3 parts Fig. 1 shows overall architecture of StepMate :

- **Piezoelectric chip** the main sensing modality of StepMate is the piezoelectric chip. Piezoelectric produces voltage whenever pressure is applied to it. StepMate utilizes this property to find the frequency of the toe movement.
- **Interfacing the micro controller** piezoelectric chip produces anolog output which cannot be used directly to infer about the physical activity. StepMate utilizes esp32 micro controller to convert the analog value into digital value and send this data to an android application for monitoring
- **Android application** After getting the data from the micro controller, the android app uses this data to classify it into different categories like whether the person is walking or running etc.

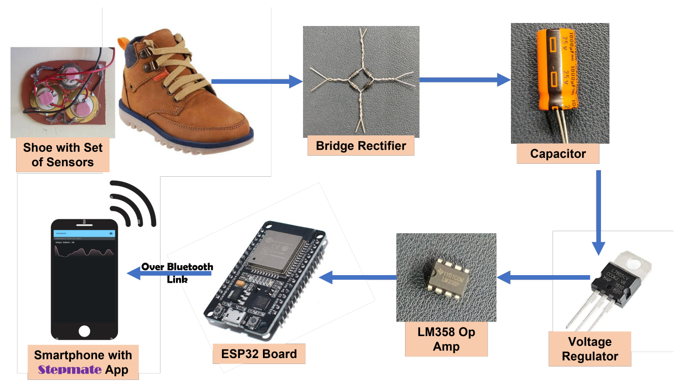

Fig. 1. StepMate architecture

A. Piezoelectric chip

Piezoelectric chip is a low cost material that produces voltage whenever pressure is applied on it. It is based on the piezo electricity property and has been widely used for energy harvesting system. However, recently its application has been shown in various other fields. Our work is very much related with the Google ATAP team shoe based sensing project. However StepMate differs from them both in implementation and application. Fig.2 shows three piezoelectric chips that are being used by StepMate to detect how many times the foot has touched the ground.

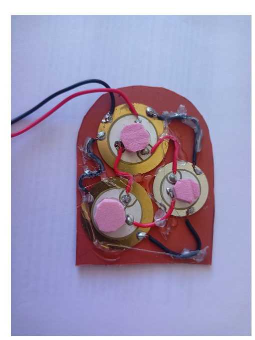

Fig. 2. Piezoelectric sensors to detect voltage spikes

Also piezoelectric chip produces AC voltage which means that the polarity of the generated voltage rotates in every cycle. Hence we have used a bridge rectifier to convert the AC voltage into DC voltage. Fig.3 shows the circuit of bridge rectifier used to convert AC to DC.

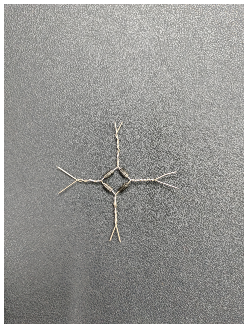

Fig. 3. Bridge rectifier for AC to DC conversion.

B. Interfacing micro controller

Piezoelectric chips produces analog data that cannot be directly used for any kind of inference. Also the converted digital data needs to be send to the mobile app for tracking. STEPMATE utilizes a esp32 micro controller for this purpose. Esp32 has both ADC and bluetooth module and hence it is used as an interface for transmitting the data. First we have attached the chips wire to the analog pins of the micro controller. Esp32 cannot read voltage above 3.3V and if high voltage is applied into it then it can damage the ADC of the micro controller. To resolve that we have stepped down the voltage using resistors to a certain level.

Fig. 4. Esp32 micro controller for interfacing

Threshold and delays are the other two parameters which play a significant role in the performance of STEPMATE . Piezoelectric chip may produce voltage spikes for very less pressure as well. These kind of voltage spikes are considered as noise and should be removed before transmitting. Also threshold should be such that it doesn't overlook the true cases. The optimal threshold should increase the true positive while reducing the false positive results. We have set a threshold by using hit and trial method. Delays is another important parameter as it decides the time interval between successive readings. We have found the optimal delay time by using hit and trial method as well.

After converting the analog data to digital data, we need to transmit it using bluetooth. We have used the bluetooth classic package to transmit the data to the mobile application. Whenever there is a spike it produces 1 or else 0. The frequency of number of ones decide the physical activity of the user.

C. Android application

The app processes the bluetooth data and displays the number of steps, current activity (IDLE / WALKING / JOGGING / SPRINTING) and activity history graph. The app was designed keeping in mind the modularity, efficiency and resiliency.

Modularity is a style of code where each functionality is divided into individual modules and they are implemented following a template. The module's implementation is its own specific but it follows the given template. If a module depends on another module, it asks the application to provide a module which follows this pattern. This way of getting an object upon which it depends is called dependency injection. This dependency injection is achieved using Dagger and Hilt libraries of android using @Inject annotation. For example, the app has an activity (this shows the UI for the user) which needs an AppViewModel object (this provides data to the UI), the app does not directly instantiate an instance of the AppViewModel but it asks the application to provide an object of the type AppViewModel. Similarly AppViewModel gets data from the BtRepository type object, which is the source of data which gets from bluetooth socket.

We added some efficiency techniques by utilizing scheduled or delayed reads from the bluetooth socket instead of using a while loop to check for data availability. We have used a delay of 1 second to get the bluetooth buffer data to process. And instead of performing these I/O operations on the main thread of the app, which can potentially hang the application, we used android provided I/O thread to perform reading from the bluetooth socket.

We have also made the app resilient where the application can withstand failures or errors by using Exception Handling. The failures are mainly due to bluetooth while reading data whether it is turned off by the system, or socket got closed, or there are no bluetooth permissions for the application. If the bluetooth is turned off, we disconnect from the device by closing the socket. If the socket gets closed unexpectedly or unavailable, we stop emitting the data and disconnect the device. If the user doesn't provide enough permissions for the app, it asks for permissions as it cannot do anything without bluetooth permissions as they are core for the working of the application.

To improve the User experience, the app stores the previously connected device information and asks the user to connect to the same device. This is also the case if the device disconnects unexpectedly due to socket error or bluetooth turned off and on.

The app does all the computation locally and doesn't transmit any private information like paired bluetooth devices list or their MAC addresses as the app doesn't need/have internet permissions.

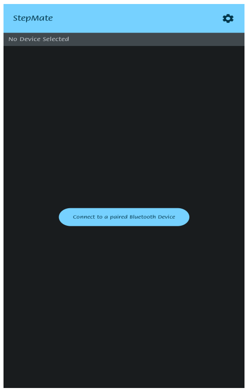

Fig. 5. Android interface

III. Results

The results obtained from the android app is shown below. We have taken four case in consideration - 1. Idle 2. Walking 3. Jogging 4. Sprinting.

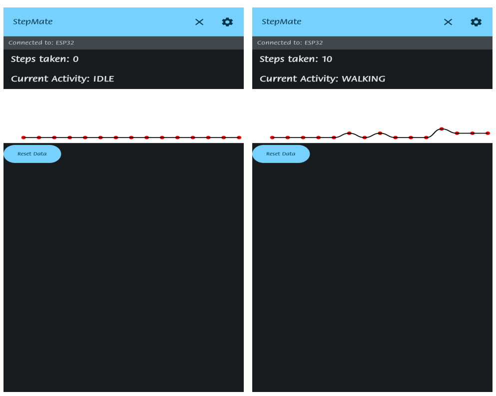

Fig. 6. Results for idle and walking condition

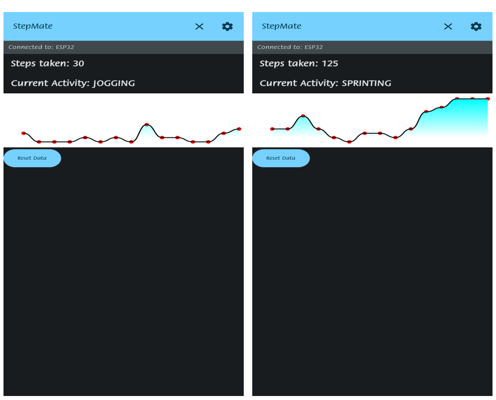

Fig. 7. Results for jogging and sprinting condition

IV. Future work

Although StepMate can provide number of steps effectively, however we want to improve it further by adding more modules into it. Walking pattern is widely used to detect heart conditions and gait anomaly detection. We want to take our work further and include these modules in StepMate . To make it more robust we want to integrate tiny ML and embedded deep learning inside StepMate . Currently StepMate consumes more power due to bluetooth classic. In our future work we want to make it more power efficient as well.

V. Conclusion

In this course project we have made StepMate which utilizes piezo electric chips for step counting. We have build the whole prototype from scratch including the android application. We have demonstrated the capability of StepMate in counting the number of steps taken by a user and the activity being done. There were some hardware limitations which we have removed by selecting optimal parameters. We hope to increase our work further and add additional modules in StepMate to make it more robust and reliable.

Video link : StepMate : Demo video

VI. References

- K. Tashiro, T. Kuroda, K. Hongo, H. Tsujimura, and Y. Naruse, "A study of step counter using piezoelectric film," in Proc. SICE Annual Conference 2010, Taipei, Taiwan, 2010, pp. 3126-3129.
- Y. Song, J. Oh, and J. Kim, "Design of a Smart Insole for Gait Monitoring Using Piezoelectric Sensors," in Proc. 6th International Conference on Smart Computing and Communications (ICSCC), 2021, pp. 1-6.
- Espressif Systems, "ESP32 Series Datasheet," 2021.
- Texas Instruments, "LM358-N Low-Power, Dual-Operational Amplifiers," 2017.
- Texas Instruments, "LM2940-N 1-A Low Dropout Regulator," 2017.
- Arduino, "Arduino core for the ESP32,"
- A. Kamavuako, E. Scheme, and K. Englehart, "Step Detection Algorithm for Piezoelectric Sensors in Prosthetic Limbs," in Proc. 42nd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC), 2020, pp. 4763-4766.
- Android Basics with Compose course — Android Developers https://developer.android.com/courses/android-basics-compose/course
- reating a graph in Jetpack Compose — by Saurabh Pant — ProAndroidDev - https://proandroiddev.com/creating-graph-in-jetpack-compose-312957b11b2