iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** XII **Month of publication:** December 2021

DOI: https://doi.org/10.22214/ijraset.2021.39585

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 9 Issue XII Dec 2021- Available at www.ijraset.com

Android Function Generator

Aayush Patel[1], Jignesh Rana[2], Abhijit Dongaonkar[3]
[1, 2]*Student,* [3]*Professor, Shri Bhagubhai Mafatlal Polytechnic Mumbai, Diploma in Information Technology*

Abstract: ***The paper focuses on an application that can be developed on android, Android is a multipurpose operating system based on Linux for mobile devices such as smartphones and tablet computers, it contains several versions such as donot, ice-cream sandwich, KitKat, pie, etc. Function generator is a device used to generate a wide range of standardized electrical pulses such as sine wave, square wave and sawtooth wave whose frequency ranges from 0.1Hz to 11,000 Hz .In this paper we aim to review how this function generator can be developed using an android mobile application. The mobile phone application uses android in order to implement a function generator which generates different A.C sources available in the laboratory. This can be used extensively in remote areas where it is not easy to carry the function generator.***
Keyword: Function Generator, Android, CRO, Signals

I. INTRODUCTION

Android is a mobile operating system based on a modified version of the Linux kernel and other open-source software, it is specially designed for touchscreen mobile devices such as smartphones and tablets.



It is free and open-source software; its source code is known as Android Open-Source Project (AOSP), which is primarily licensed under the Apache License.

The source code has been used to develop Android variants on a range of other electronic products, such as game consoles, digital cameras, media players, Personal Computers, each with a dedicated user interface. Some well-known derivatives include Android TV for TVs and Wear OS for wearable devices

Since 2011, Android has been the world's best-selling smartphone operating system, and since 2013 it has been the best-selling operating system on tablets. As of May 2021, it has more than 3 billion monthly active users, the greatest installed base of any operating system, and as of January 2021, the Google Play Store features over 3 million apps. Android 12, released on October 4, 2021, is the latest version

A. Overview of Android Studios



Android Studios is one of the well-known IDE among Android App developers, it is the official IDE for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on macOS, Windows, and Linux based operating systems. The latest version of Android Studios for macOS and Windows is Arctic Fox 3.1 and for Linux is 4.1

Android Studio supports programming languages, such as Java, Kotlin, C++, and more with extensions, all Java 7 language features and a subset of Java 8 language features that vary by platform version. However, IntelliJ states that AS is compatible with all released Java versions, and Java 12, it is not clear to what level Android Studio supports Java versions up to Java 12.

Once an application has been built with Android Studio, it can be published on the Google Play Store. The application must comply with the content policy for developers of the Google Play Store.

B. Overview of Flutter



Flutter is an open-source UI software development kit created by Google. It is used to build cross platform apps for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase. The initial version released was Alpha (v0.06) in 2017. Its stable version released was Alpha (2.5) in 8th September 2021. Flutter supports programming languages, such as Dart, C, C++.[5]

©IJRASET: All Rights are Reserved

Advantages of Flutter
Faster Coding
Single code for Android and IOS
Less testing
Same app UI older devices
Reduce code development time
Designs that your user will love
Flutter coding
Increased Time-to-Market Speed
Perfect for Minimum Viable Product (MVP)

Advantages of Android Studios
Instant Run
Android Emulator
Cloud Test Lab
GPU Debugger Preview
App Indexing

This paper consists of the following sections –

1) Introduction
2) Literature survey
3) Android development languages
4) Current research
5) Conclusion

II. LITERATURE SURVEY

Android mobile app development has benefited to mobile phone users to do several tasks efficiently such as sending a mail, catch a cab or attend online lectures due to Covid.

In the paper “Mobile Function Generator Using Android” authors Elijah Mathew, Dr. Sunil Karamchandani talks about how we can use mobile app to develop sin, square and triangle waves that is substitute of function generator to be used in remote location

III. ANDROID DEVELOPMENT LANGUAGES

A. Java

Java was the official language for Android App Development. Most apps in the Play Store are built with Java, and it is also the most supported language by Google. Java also has a great online community for support in case of any bugs.

Java is a baffling language for a beginner to use as it contains complex topics like constructors, null pointer exceptions, concurrency, checked exceptions, etc.

B. Kotlin

Kotlin is the official language for Android App Development declared by Google since 2019. Kotlin is a cross platform language that may be used as an alternative to Java for Android App Development. Kotlin can interoperate with Java and it runs on the Java Virtual Machine.

C. C++

C ++ can be used for Android application development using the Native Android Development Kit (NDK). However, an application cannot be fully created using C ++ and the NDK is used to implement parts of the application in native C ++ code. This helps to use C ++ code libraries for the application as needed.

While C ++ is useful for Android application development in some cases, it is much more difficult to configure and is much less flexible. It can also lead to more errors due to increased complexity. Therefore, it is better to use Java compared to C ++ as it does not provide enough gain to compensate for the efforts required.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 9 Issue XII Dec 2021- Available at www.ijraset.com

IV. CURRENT RESEARCH

A. *Project Overview*

Function generators are bulkier, heavy and not feasible to carry everywhere. Already several researchers are working onto developing a function generator's signal using a mobile application and transmit it to the other device using the AUX port on android devices. This signal generation involves creating sine, triangle or square wave individually one at a time. This application will help us get the signals even at remote locations where we cannot carry the gigantic function generators.

Every technology comes with certain limitations like with mobile application we can generate the signals, but its output voltage is too low and low bandwidth (can only generate signals within 20 Hz to 20,000 Hz). Another drawback is that at most 2-3 volts output is generated.

B. *Procedure*

Our Research mainly consists of 3 phases:

1) Generating sin, sawtooth and square waveforms of varying frequencies using java.
2) Then these generated waveforms need to be transmitted to the other device using AUX port using Android capabilities and hardware along with Android and java coding.
3) The waveforms are now available at the device connected with AUX port then need to be interfaced to the CRO or the working project.

C. *Snippet for the Waveform Generation using Variable Input*

```
new Thread( new Runnable( )
    {
      public void run( )
      {
        final float frequency = 10000;
        float increment = (float)(2*Math.PI) * frequency / 44100;
        // angular increment for each sample
        float angle = 0;
        AndroidAudioDevice device = new AndroidAudioDevice( );
        float samples[] = new float[1024];
        while( true )
        {
          for( int i = 0; i < samples.length; i++ )
          {
            samples[i] = (float)(Math.sin( angle ));
            angle += increment;
          }
          device.writeSamples(samples );
        }
      }
    } ).start();
```

[1]

This is the code to create a thread and generate a sign wave pulse of constant frequency of 10 KHz. To take variable frequency input we need to create a digital potentiometer knob either vertically or horizontally on the mobile screen with frequency of minimum 20 Hz to 20000 Hz. We can use the length of the variable potentiometer knob and use the scale function to compute the frequency input. For example, the length of the knob is 5 cm, and the user places the knob at the exact half that is 2.5 cm which will give us the frequency of 10010 Hz.

©IJRASET: All Rights are Reserved

D. Interface with CRO

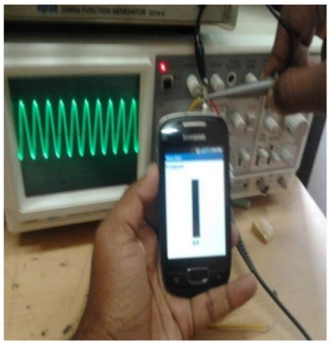
To generate the output pulse using the audio (aux) port in the android mobile we first need to understand the structure of the aux port. The 3.5 mm aux port is made of 4 parts left speaker +, right speaker +, mic and ground. [1]

To generate the signal in the 3.5mm Aux port we need to convert the digital signal into an analog signal. For this we can use the popular java class *AudioTrack* to play some audio resources (in our case, the waveform) from java application [6]



An AudioTrack instance works on 2 modes:

1) *Streaming:* Here the application writes a continuous stream of data to the AudioTrack, using one of the write() methods. These are blocking and return when the data has been transferred from the Java layer to the native layer and queued for playback. [6]
2) *Static:* This should be chosen when dealing with short sounds that fit in memory and that need to be played with the smallest latency possible. The static mode will therefore be preferred for UI and game sounds that are played often, and with the smallest overhead possible.

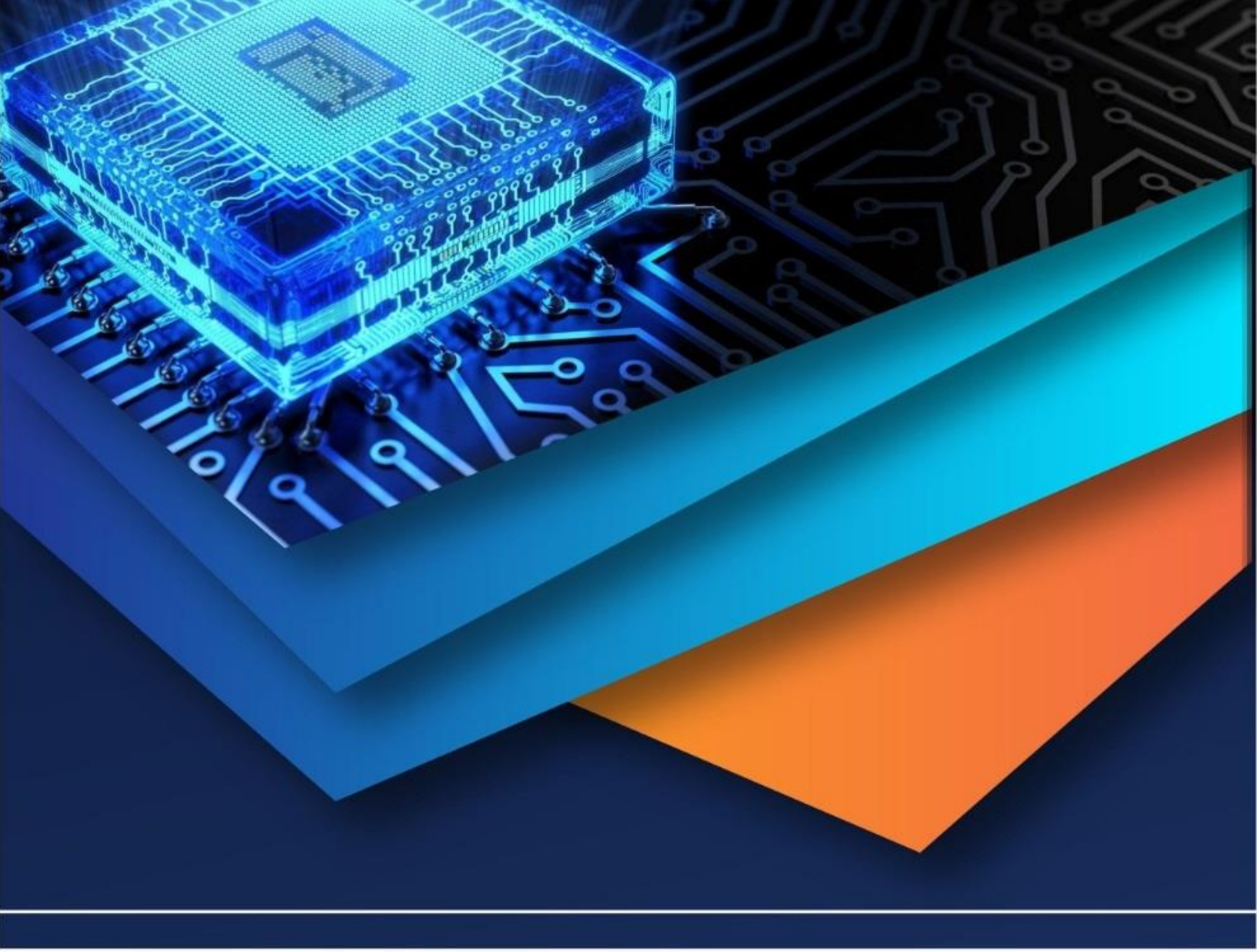

[1]

V. CONCLUSION

Thus, we have researched on an application using android implementing a function generator on a cell phone of which the sine wave generated is displayed above. It is indeed a great advantage when function generators are unavailable at home or at work field so we reviewed about how to build mobile app which could transmit signals like function generators using AUX port on the android mobile devices. It is dominant to provide a function generator application into hands of every researcher possessing a smart phone. Can use this app in isolated places. Now we don't have to carry heavy and bulky function generator everywhere. It will save the cost of investment by the user to buy a function generator for small applications. Taking power and other considerations into account, there will be no need for any function generators in the lab for years to come, as everyone will have function generators in their phone.

REFERENCES

[1] Elijah Mathew Mobile Function Generator Using Android, 2015, D. J. Sanghvi College of Engineering, Mumbai
[2] IBM (International Business Machines) Arvind Krishna CEO of IBM https://developer.ibm.com/articles/os-android-devel/
[3] Britannica Jorge Cauz CEO of Britannica https://www.britannica.com/technology/Linux
[4] https://code.tutsplus.com/articles/android-from-scratch-an-overview-of-android-application-development--cms-25972
[5] https://thebroodle.com/android/flutter-vs-android-studio-which-is-better/
[6] https://developer.android.com/reference/android/media/AudioTrack



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)