

Wireless Oscilloscope Powered By Android

Bhagyashree D. Hatwar¹, Prof. Amol C.Wani²

Electronics & Telecommunication Dept., S.S.B.T's COET, Bambhori, Maharashtra, India

Abstract-- Oscilloscopes which are available may be CRO or DSO is not handy, user-friendly, not portable, and requires more power consumption. If we are doing any analysis regarding any circuit, then to see its synthesis we move towards oscilloscope. So, if oscilloscope can be handy with the help of android as an operating system, whenever we want to see our desirable waveform, it is in our hand. We are developing the real time application of android oscilloscope. This paper presents the design and implementation of a low cost, portable, light-weight; low power, dual-channel oscilloscope, consisting of a hardware device and a software application. The device is equipped with a Bluetooth module to provide connectivity to a device with Bluetooth, running the Android operating system (OS), in order to display the waveforms.

Keywords-- Android, Oscilloscope, Bluetooth

I. INTRODUCTION

Portable oscilloscopes currently in the market are very expensive, less power efficient and have small low resolution displays. This paper presents the design and implementation of a low cost, portable, light-weight, low power, dual-channel oscilloscope, consisting of a hardware device and a software application. The device is equipped with a Bluetooth module to provide connectivity to a device with Bluetooth, running the Android operating system (OS), in order to display the waveforms. Android OS is selected because there are a decent number of Android device users and most of these devices satisfy the requirements of the oscilloscope's software application. The hardware device includes circuitry to capture the input voltage signals and an embedded Bluetooth module for transmitting the captured signal information to an Android device for displaying the waveform. The Software application developed for Android receives the data transmitted from the hardware device and plots the waveform according to the display settings configured by the user. These display configurations are transmitted to the hardware device once they are set by the user, and are used by the hardware device to set the sampling rate and the values of samples. For optimal use of the available bandwidth, the application provides single mode of operation, namely single channel mode where only channel 1 or channel 2 is operational.

The user can select a mode from the application, which in turn sends a message to the microcontroller which then changes the sampling frequency accordingly same sampling rate for channel 1 and for another channel 2 [1].

II. LITERATURE SURVEY

The implementation of an oscilloscope with Bluetooth was previously reported, by Yus in 2010 [2]. It is an open source prototype project called the "Android Bluetooth Oscilloscope", which consisted of a Bluetooth enabled transmitter circuit to send data to an Android phone which draws the waveforms on its screen. The transmitter circuit uses Microchip's dsPIC33FJ16GS504 and an LMX9838 Bluetooth 2.0 SPP module. The maximum input voltage to the circuit is +8 V to -8 V. However, there is no mention about the bandwidth of the device. Furthermore, it is stated that the application had been tested only with a Samsung Galaxy GT-i5700 Spica (rooted Android 2.1 OS) phone [1].

III. SYSTEM OVERVIEW

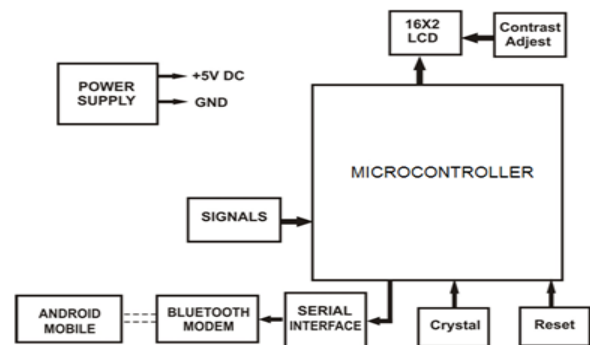


Fig.3.1. Block diagram of Overall System

The Hardware: The block diagram of overall system is as shown in fig.3.1. The Atmega16 is a low-power CMOS 8-Bit microcontroller based on the AVR enhanced RISC Architecture. By executing powerful instructions in a single clock cycle, the Atmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designed to optimize power consumption versus processing speed. The device is manufactured using Atmel's high density nonvolatile memory technology.

The Atmel ATmega16 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The transmitter circuit uses Microchip's Atmega 16 for the analog-to-digital conversion of the input signals on two channels. The processed data on the Atmega 16 are then transmitted to the phone (for waveform display) via the Bluetooth HCI module.

Bluetooth Wireless technology is a short range communications technology intended to replace the cables connecting portable or fixed devices. It transmits data via low power radio waves. Bluetooth technology has the ability to handle both data and voice transmissions simultaneously. The key features of the Bluetooth technology are robustness, low power, and low cost.

From research carried out it was found that data rates of 2 Mbps are not achievable with the existing software stacks implemented on the module's controller. Therefore, the approach suggested to fully utilize the Bluetooth bandwidth, was to use the module in Host Controller Interface (HCI) mode.

The standard mode refers to the on-board stack running on the Bluetooth module. In HCI mode, the on-board stack is bypassed and the module is put in a state that runs the Bluetooth baseband. Therefore, the Bluetooth stack is not on the module, so it needs to be implemented on the interfacing host processor. So the module is used as a radio, performing the lower level MAC functionalities, while the application stack runs on the host processor. The hardware interface between the host processor and the module can be either UART or USB. The advantage of HCI mode is that it allows to obtain maximum throughput and also to implement custom profiles on the Bluetooth module. The Android device communicates in the RFCOMM layer [3] using the SPP profile, but in HCI mode it will have to work in the lower baseband layers. The support currently available for such an approach in Android is inadequate. Fig.3.2.shows the diagram of HC-06.

The proper selection for our work is Bluetooth module HC06, because of following features:

Master and slave mode can't be switched.

Master role: It has no function to remember the last paired slave device. It can be made paired to any slave device.

Pairing: The master device can not only make pair with the specified Bluetooth address, like cell-phone, computer adapter, slave device, but also can search and make pair with the slave device automatically.

Multi-device communication: There is only point to point communication for modules, but the adapter can communicate with multi-modules.

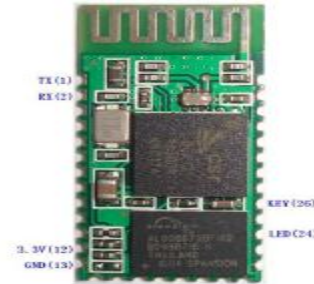


Fig.3.2. Diagram of HC-06[7]

Basic Specifications of the Bluetooth Embedded Device

1. Number of input channels: 2
2. Bandwidth: 1 kHz (single channel and dual channel mode)
3. Input voltage range: -5V with 1:1 probe
4. Sampling frequency: 5 kHz (single channel and dual channel mode)

IV. BLUETOOTH CONNECTIVITY

The Android platform includes support for the Bluetooth network stack, which allows a device to wirelessly exchange data with other Bluetooth devices. The application framework provides access to the Bluetooth functionality through the Android Bluetooth APIs. These APIs let applications wirelessly connect to other Bluetooth devices, enabling point-to-point and multipoint wireless features. Fig.4.1.shows Bluetooth permission request. Using the Bluetooth APIs, an Android application can perform the following:

- Scan for other Bluetooth devices
- Query the local Bluetooth adapter for paired bluetooth devices
- Establish RFCOMM channels
- Connect to other devices through service discovery
- Transfer data to and from other devices
- Manage multiple connections

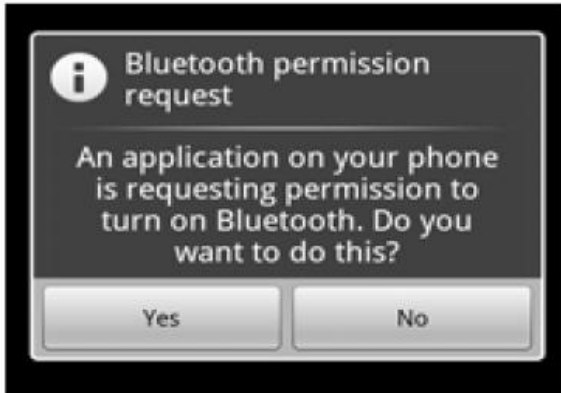


Fig.4.1. Bluetooth Connectivity

V. ANDROID SMARTPHONE

Android is an operating system based on the Linux kernel, and designed primarily for touch screen mobile devices such as smartphone and tablet computers. Initially developed by Android, Inc., which Google backed financially and later bought in 2005, Android was unveiled in 2007 along with the founding of the Open Handset Alliance a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. The first publicly available smartphone running Android, the HTC Dream, was released on October 22, 2008.

The user interface of Android is based on direct manipulation, using touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching, and reverse pinching to manipulate on-screen objects.

Android is hailed as “the first complete, open, and free mobile platform.”[5]

i) Complete: The designers took a comprehensive approach when they developed the Android platform.

They began with a secure operating system and built a robust software framework on top that allows for rich application development opportunities.

ii) Open: The Android platform is provided through open source licensing. Developers have unprecedented access to the handset features when developing applications.

iii) Free: Android applications are free to develop. There are no licensing or royalty fees to develop on the platform. No required membership fees. No required testing fees. No required signing or certification fees. Android applications can be distributed and commercialized in a variety of ways [8]. Fig.5.1.shows architecture of android.

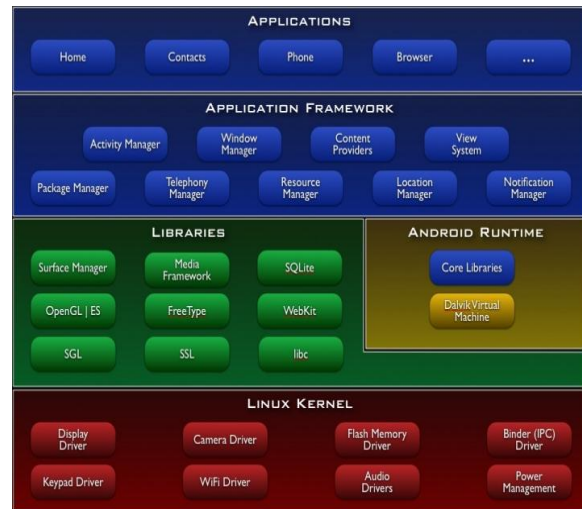


Fig.5.1. Android Architecture [6]

Android Application

The Android application is developed based on the software development kit (SDK) of Android 2.2 Froyo OS, API (Application Program interface) level 8.

The Android device used for testing the application is a Sony Xperia M smartphone with Android OS, V4.1 (Jelly Bean) with Dual core 1GHz krait. The smartphone has a TFT capacitive touch with 480 x 854 LCD screen with Bluetooth v4.0 + HS (High Speed). Fig.5.2. shows the android mobile which is used for experimental purpose. Fig.5,3, shows layout of the application on the phone’s screen.



Fig.5.2. Android mobile Sony Xperia M [9]

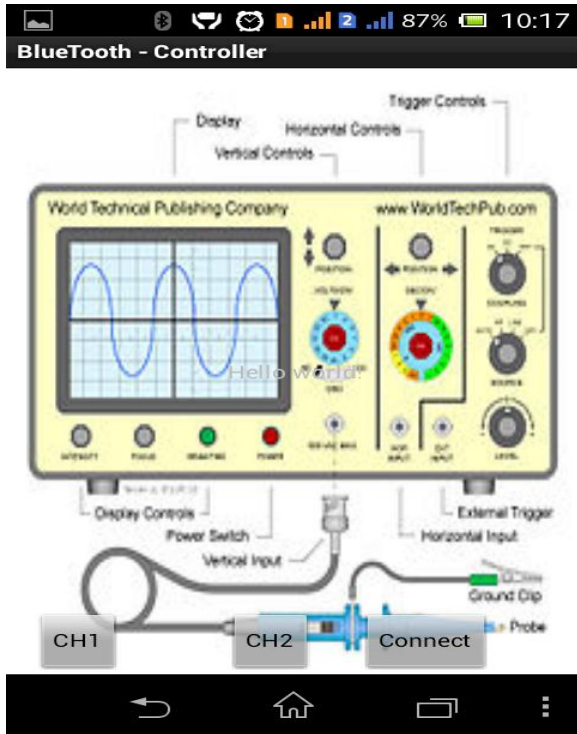


Fig.5.2. Layout of Android Oscilloscope

VI. WAVEFORM PLOTTING ALGORITHM

The transmission format in Figure 6 also makes the reception of data easier. The application reads the data stream from its Bluetooth data buffer and separates out the channel 1 and channel 2 data into two arrays. Reading this continuous data stream with the un-rooted Android OS and synchronizing with the transmission rate was the major challenge in developing the application. Thereafter, the waveform of the particular channel is plotted on the grid by joining the data points (samples) together with straight lines. [1]

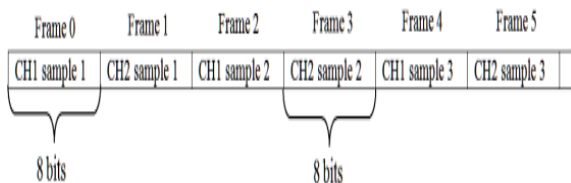


Fig.6. Data transmission sequence in single channel mode[1]

VII. MODES OF OPERATION

The device provides one mode of operation, namely single channel mode where only channel 1 is operational, with the full bandwidth available for channel 1. Similarly for channel 2 also full bandwidth available. The sampling frequency in single channel mode is 5 kHz. The user can select a mode from the application, which in turn sends a message to the Atmega16 to notify the selected mode, so it can change the sampling frequency. This helps to make efficient use of the available bandwidth when only one channel is required.

By default, the hardware is in single channel mode, so both input channels are not sampled simultaneously. After sampling and conversion, Channel 1 (CH1) and Channel 2 (CH2) samples are converted to a byte and are immediately written to the micro-controller's UART buffer for transmission via Bluetooth. Finally depending on sampling frequency the waveforms are displayed on Android Oscilloscope.

VIII. RESULTS

The operation of the device was tested by providing test signals from the signal generator and ac voltages from the ac power supply. Both modes of operation of the device were tested with different voltages and frequencies applied to the channels. Figure 7.1 and 7.2 shows screen shots of the application displaying the waveforms of various test conditions.

The contribution towards the paper is creating the algorithm on Android SDK. Applications are usually developed in the Java programming language using the Android Software Development Kit.

The Android software development kit (SDK) includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, Windows XP or later; for the moment one can develop Android software on Android itself by using [AIDE - Android IDE - Java, C++] app and [Android java editor] app. The officially supported integrated development environment (IDE) is Eclipse using the Android Development Tools (ADT) Plugin.

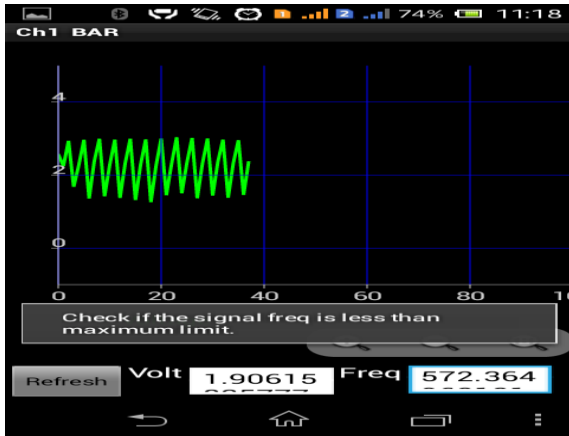


Fig.7.1.Triangular wave

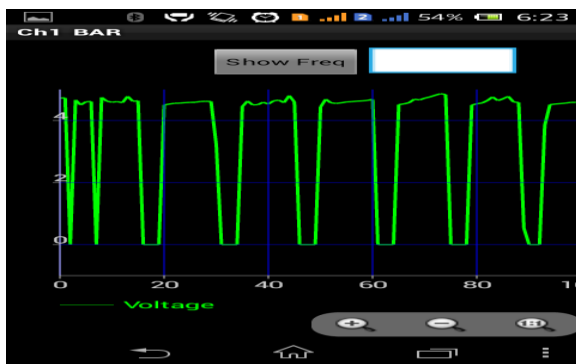


Fig.7.2. Square Wave

IX. COMPARISON OF CRO, DSO AND ANDROID OSCILLOSCOPE

Table 8.1.
Comparison of CRO, DSO and Android Oscilloscope

PARAMETERS	CRO	DSO	ANDROID OSCILLOSCOPE
DESIGN	COMPLEX	COMPLEX	SIMPLE
FREQUENCY RESPONSE	AC:1Hz to 1MHz	AC:10Hz to 200MHz	AC:0 to 5KHz
ACCURACY	+1% to -1%	+3% to -3%	+1 to -1%
STORAGE TIME	NO SPACE TO STORE	FIXED INTERVAL OF TIME	USING SCREEN SHOTS
PORTABILITY	NO	NO	YES
WEIGHT	BULKY (7.2Kg)	MODERATE BULKY (2.5Kg)	LIGHT-WEIGHT (500g)
COST	MODERATE (26,000)	HIGH (35,000 - 60,000)	LESS (5400/-)

The above table shows the comparison of oscilloscopes which are available in market like CRO, DSO with our android oscilloscope. All above mentioned values of CRO, DSO and Android Oscilloscope are approximate values. Actually we are implementing the idea of CRO on android mobile. So we are focusing on parameters like portability, power consumption and cost. Future implementation may also result in FFT, Spectrum Analyzer etc. all the features that are available on DSO or CRO can be possible in our android oscilloscope.

X. CONCLUSION

This paper conveys the basic idea related to low-cost, portable, low-complexity Bluetooth embedded oscilloscope.

The hardware device includes circuitry to capture the input voltage signals and an embedded Bluetooth module for transmitting the captured signal information to an Android device for displaying the waveform. The Software application developed for Android receives the data transmitted from the hardware device and plots the waveform according to the display settings which is set by the user. Using Bluetooth module, higher data transmission is possible. Android based smartphone is most popular now. Android has emerged as a new mobile development platform, building on past successes and avoiding past failures of other platforms. Android was designed to empower the developer to write innovative applications. So, it can become a portable oscilloscope. Also we have actually implemented this work in real time applications. We have tested this system in basic circuits like integrator, differentiator. Using further advancement, FFT can be obtained. Thus android based oscilloscope plays a vital role in the field of research.

REFERENCES

- [1] H.M.D.B. Seneviratn and K.N. Abhayasingh, "Bluetooth Embedded Portable Oscilloscope" SAIMT Research Symposium on Engineering Advancements 2013 [SAITM – RSEA 2013].
- [2] Yus "Android Bluetooth Oscilloscope" (2010, Sep. 23).
- [3] Android Developers, "Bluetooth Socket", Mar.5, 2013.
- [4] Martin Jannus and Andres Udal "USB Audio Chip Based Oscilloscope and Signal Generator for Mobile Laboratories" (June 2011).
- [5] Roland Szabo, Aurel Gontean, Ioan Lie, and Mircea BabaiNa "Oscilloscope control with PC" International Journal of computer and communication, Issue 3, volume 3, 2009.
- [6] Kalyani Ganvir and Hemant Waghmare, "Oscilloscope on Android Phone" Proceedings of IRAJ International Conference, 21 July 2013.
- [7] HC bluetooth module user manual datasheet.
- [8] Android Development. Online available <http://developer.android.com>
- [9] Sony Xperia M Specifications from GSM Arena