Journal of the Nigerian Association of Mathematical Physics
***Volume27*, (*July, 2014*), pp 405 – 412**
© J. of NAMP

Implementation of a 20Khz Bandwidth Portable Oscilloscope

Bello N and Osarogiagbon A.U.

Department of Electrical/Electronic Engineering, University of Benin, Benin City.

Abstract

Oscilloscope is a vital tool in electronic engineering but it is not common for hobbyist because of its cost.We decided to design and implement a very simple one that is both portable and self contained in terms of power and functionality, using off-the-shelf components and can still provide comparable performance with the professional and commercial ones. In the design and construction of the oscilloscope, PIC18F4620was programmed to acquire, sample and process analog test signals and a 128x64 Graphic Liquid Crystal Display (GLCD) driven by the famous Samsung KS0108 GLCD driver was used to draw and display the waveform of the processed signal. A program was written in C programming language for the PIC18F4620. This program implements the functionality of sampling analog input signal at the required Nyquist sampling rate for 20 KHz bandwidth and also present math function of input signal. Labcenter Proteus was used to simulate and test the functionality and a Development board (easyPIC6) was also used. On completion, a prototype of a 20 KHz oscilloscope with math function was realized with a satisfactory functionality.

Keywords: Sampling, Discrete signal, periodic, Nyquist Theorem, Analog to Digital converter (ADC), Filter, Graphical Liquid Crystal Display (GLCD), Sample and Hold, Aliasing

1.0 Introduction

An oscilloscope, previously called an oscillograph, and informally known as a scope, CRO (for cathode-ray oscilloscope), or DSO (digital storage oscilloscope), is a type of electronic test instrument that allows observation of constantly varying signal voltages, usually as a two-dimensional graph of one or more electrical potential differences using the vertical or 'Y' axis, plotted as a function of time (horizontal or 'x' axis). Many signals, for example sound, can be converted to voltages and displayed this way. Most of the signals in existence are often periodic and repeat constantly, so that multiple samples of a signal which is actually varying with time are displayed as a steady graph. Oscilloscopes are commonly used to observe the exact waveform of an electrical signal.

1.1 Oscilloscope Design Objectives

We look into a detailed design and construction of a simple and cheap digital oscilloscope with a few functions: signal storage and retrieval, mathematical functions implementation and a way to strobe the display. The sampling rate of this simple oscilloscope exceeded 50 KHz to 70 KHz. The design of a cheap digital oscilloscope also gave a general view about the concept of analog signal sampling, conversion and analysis in the field of electrical/electronic engineering and a review on Nyquist sampling theorem.

1.2 Sampling

Sampling is the process of converting a portion of an analog input signal into a number of discrete electrical values for the purpose of storage, processing and/or display [1]. The magnitude of each sampled point is equal to the amplitude of the input signal at the instant in time in which the signal is sampled.

Sampling is like taking snapshots. Each snapshot corresponds to a specific point in time on the waveform. These snapshots can then be arranged in the appropriate order in time so as to reconstruct the input signal [1].

1.2.1 Sampling Rate

Sampling rate is the rate at which sampling occurs in digital signal processing. It is often used to rate a digital oscilloscope [2]. For a successful design of a digital oscilloscope, knowledge of Nyquist sampling criterion is needed.

Corresponding author: Bello N E-mail: nosabello@yahoo.com, Tel.: +2348025373737

The Nyquist sampling theorem provides a prescription for the nominal sampling interval required to avoid aliasing. It may be stated simply as follows: The sampling frequency should be at least twice the highest frequency contained in the signal[3].

Mathematically, $f_s \geq 2f_c$ -- (1)

Where f_s is the sampling frequency (how often samples are taken per unit of time orSpace), and f_c is the highest frequency contained in the signal. The effect of not meeting the Nyquist sampling criterion results to aliasing. Aliasing arises when a signal is discretely sampled at a rate that is insufficient to capture the changes in the signal which causes loss of information in the signal [4].

1.3 BASIC STRUCTURE OF DIGITAL OSCILLOSCOPE DESIGN

Fig. 1 gives a general structure of a basic digital oscilloscope. It consists majorly of a low-pass filter and digital signal processing hardware implemented in advanced processor level or microcontroller level and a dedicated display hardware.

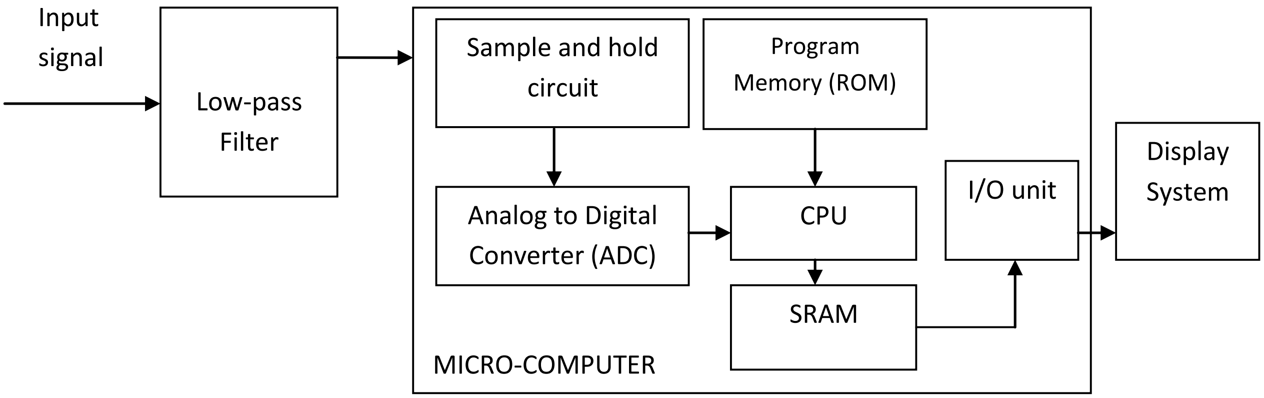

Fig. 1: Basic Digital Oscilloscope Structure

A low-pass filter may be required in an oscilloscope to remove unwanted frequency above the maximum from the test signal. This frequency may be as a result of electrical noises. Since the analog to digital conversion (ADC) process takes a finite amount of time to complete, the Filtered signal is fed through a sample and hold circuit prior to the ADC [5]. This step ensures that the voltage at the input of the ADC is constant over the duration of the conversion process. After a successful conversion of sampled analog signal to digital, the CPU under the control of a written instruction stores sampled data in Memory typically in a Static RAM where it can easily be processed according to the program stored in the program memory. Such processes include the necessary algorithm used to convert sampled signals to presentable and useful information that can be displayed on a digital screen.

1.4 Previous and Related Digital Oscilloscope Designs

There exist several methods of digital oscilloscope design such as Digital storage oscilloscope, Digital sampling oscilloscopes, Cathode-ray oscilloscope (CRO), Dual-beam oscilloscope e.t.c. [6]. Never the less, each method of digital oscilloscope design operates using the same principle, which can be described as a chained process of: feeding analog signal into an Analog to Digital converter (ADC), obtaining the Digital output representation of the (n) sampled analog signal and a readable or understandable interpretation and representation of the signal wave form, using a suitable visual display technology. Differences exist by the way each method processes and analyses its acquired signal and a proper way of drawing the output waveform on the screen. It was observed (from here) that, the type of display technology affects the overall design of a digital oscilloscope. We take a look at previous design methods.

1.4.1 Solid State Oscilloscope

Previous works, “Solid state oscilloscope Development” by Orukpe [7] and Katz [8], used Light Emitting Diode (LED) for their display, which was an array of LED matrix to display the wave form of a processed input signal. The design method incorporates simple analog to digital converter, logic gates and switches to draw the pattern of a signal waveform, by addressing corresponding LED with X and Y values. The ADC which accepts an analog signal and produces a digital output generates the Y-value and a decade counter switches along the X axis to provide the X-value. Though the design method proved its success as a digital oscilloscope, it was not flexible. Besides, it was huge and it consumes much power due to the array of LEDs used. Also, it was observed theoretically that the power consumption would increase noticeably, if it is desired to increase the resolution of the LED array.

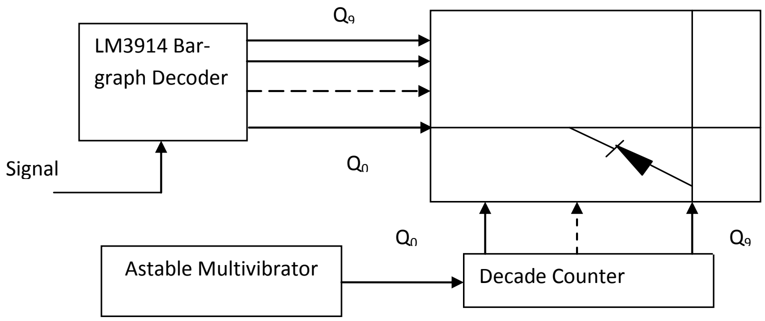

Fig 2: Typical 10x10 LED Array Solid state Oscilloscope Schematic

1.4.2 PC Scope

This method uses an electronic hardware module to acquire and store samples of analog signal into memory. After which it requires the use of a Personal computer over a convenient interface (RS232 or USB) to read the stored samples from the hardware for processing and to display signal waveform on the computer screen. Bulk of the signal processing is done on the PC sideunder the control of a special and dedicated program [9].This method discourages flexibility and ease of mobility, and does not exist as a unit because it would always require the use of a Personal Computer, therefore increases the cost of implementation and maintenance.

2.0 Approach and Methodology

Considering related works and their cost, we designed and implemented a portable digital oscilloscope with the intension of making it a mobile and pocket size one, using of-the-shelf components. In implementingthe oscilloscope, we used a PIC184620 which features an inbuilt analog to digital converter (ADC) module, several Input and Output ports, with 4K of static RAM and 1024 bytes of EEPROM. We wrote a program in C Programming language to carry out the configuration of respectiveInput/Output (I/O) pins that were used as Analog and Digital I/O. Next,we created functions to process the stored samples.Such processes included the implementation of math functions which are: Maximum Value, Minimum Value, signal Peak to Peak value and Average value. Finally we used some compiler built-in functions to format and draw signal wave form representation on a 128 x 64 GLCD screen. A basic flowchart of the implemented Digital Oscilloscope is shows in Fig.3.

3.0 Design and Analysis of Digital Oscilloscope

The block diagram of the 20 KHz Digital oscilloscope is shown in Fig.4, implemented using PIC18F4620. In the course of design a carefully selected probe was used to connect test signal to the oscilloscope. This was done because according to design specification and accuracy a wrong probe might introduce noise into a test signal that has a high frequency close 15 KHz. For this reason a correct choice of probe was considered.

Microcontroller used, (PIC18F4620) requires an oscillator to ensure the stability of the selected clock frequency in the configuration bit of the microcontroller. For correct Nyquist sampling rate, more accuracy and surer result, we decided to use an external clock. We chose 20MHz crystal clock for the microcontroller to achieve fastest possible speed while still staying within the microcontroller's acceptable oscillator frequency range. The 20MHz oscillator provided the microcontroller program to run at 5 Million instructions per seconds (MIPS). Fig.4 shows the circuit diagram of the 20 KHz portable oscilloscope.

Themicrocontroller was connected to the graphical LCD (GLCD) through PORTD I/O pins of the microcontroller to provide data and the appropriate control signals as shown in Fig.5. The GLCD hosted an embedded driver chip KS0108 which functioned to drive an LCD display glass of 128 columns by 64 rows of pixels under the supply of 5VDC. In this design the GLCD was used to display and visualize the processed sampled waveform, math functions and other information of the test signal.

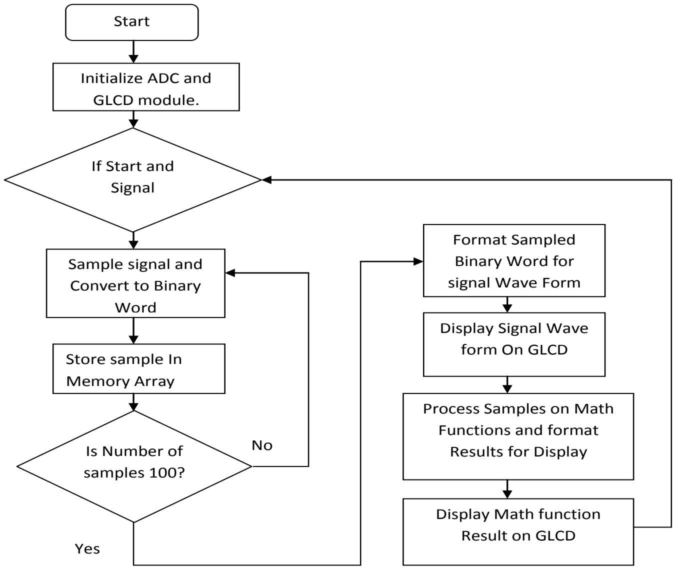

Fig 3: Flowchart of Implemented Digital Oscilloscope

To achieve some useful functions like RECALL, SAVE and STROBE we included a group of tic-tac push buttons connected from pin 0 through pin 2 (RC0, RC1, RC2) on the PORTC of the microcontroller; it was made from three micro-switches with 10kΩ pull-down resistors connected to each of them. This configuration placed the three pins on the PORTC at logic zero. When a button (micro-switch) is pressed, it sends logic 1 into the respective pin and this is then processed by the application code running on the microcontroller.

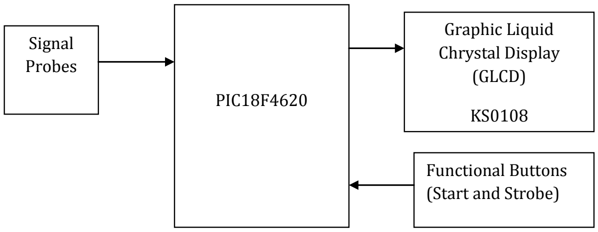

Fig 4: Block Diagram of Implemented Digital Oscilloscope

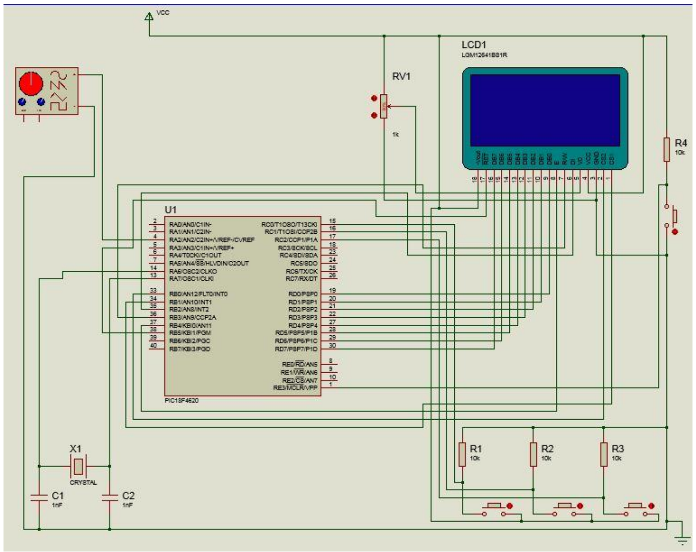

Fig 5: Schematic of Digital Oscilloscope

3.1 Microcontroller Source Code (Written In C)

The code that runs in the microcontroller to process/analyze and format the signal for the display hardware was written and developed in C programming language. The compiler/IDE used for the development of the microcontroller application source code was MikroC Pro™ from MikroElektronika®. The generated .hex file was transferred to the microcontroller using a PIC development board (EasyPIC6) from MikroElektronika®, which functioned as a PIC microcontroller hardware programmer and a testing tool.

3.2 How the C Code Works

For a successful operation of the proposed oscilloscope, we decided to sample the analog signal from Pin RA2 of the microcontroller at the Nyquist sampling rate, since the oscilloscope had been chosen to operate in a bandwidth of 20 KHz. Therefore to satisfy the Nyquist sampling theorem, our sampling rate in-code was twice the chosen bandwidth which gave 40 KHz; PIC18F4620 has an inbuilt 13 channel, 10 bits analog to digital converter module. This makes it capable of producing a 10 bit number representing the sampled analog signal, thus providing:

$2^{10} = 1024 \text{ levels of discritization}$ -- (2)

The microcontroller's A/D converter module was configured in-code by setting some special function registers (SFR) of the microcontroller (i.e. the ADCON0, ADCON1 and ADCON2 registers); the binary number representation of the sampled analog wave was fetched from the ADRESH (Result High Register) and ADRESL (Result Low Register) of PIC18F4620. We took hundred samples of the analog signal per time and stored their binary number representation in an array for further processing to enable us display them properly on the GLCD. For a proper drawing of signal wave form on the GLCD, All the values calculated were scaled down by multiplying the sampled signal's amplitude by voltage factors so that they can successfully be displayed on the 128 by 64 GLCD screen. The following general formula was used;

$\text{Voltage scale factor} = \frac{\text{voltage range}}{\text{height of GLCD display area}}$ ------------------------------ (3)

The GLCD display height will range from 0 to 63; this will place the height (max. value) at 63. For voltage range of 0 to 5 volts;

$\text{Voltage scale factor} = \frac{5}{63} = 0.0793651$ -------------------------------- (4)

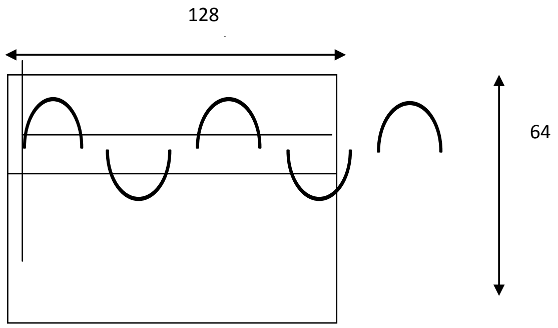

Fig.6: GLCD pixel resolution

3.3 Math Function Implementation and Techniques

The maximum number that can ever be generated by the ADC module is $2^{10} = 1024$ (in decimal). Therefore the ADC resolution is given by:

$\text{Voltage division (resolution)} = \frac{\text{Maximum input voltage}}{1024}$ -------------------------------- (5)

In code, the following math functions on input signal were performed:

I. Maximum Value
II. Minimum Voltage
III. Peak to Peak Voltage
IV. Average Value
V. Frequency

Maximum Voltage:

The maximum voltage is calculated as follows:

$\text{Max. voltage} = \text{max. binary number} \times \text{voltage division (resolution) of the ADC}$ -------(6)

The Minimum Voltage:

Minimum Voltage value of the signal was calculated from the hundred samples in the same way we calculated for the maximum voltage value, i.e.

$$\text{Min. voltage} = \text{min. binary number} \times \text{voltage division (resolution) of the ADC} \quad \text{-----------} \; (7)$$

Peak to Peak Value:

The peak to peak value of the analog signal was calculated by subtracting its minimum sampled value from its maximum value and multiplying all that with the voltage division to get the actual value in volts.

$$V_{p-p} = (\text{max. binary number} - \text{min. binary number}) \times \text{voltage resolution} \quad \text{------------------} \; (8)$$

Average Value:

The average value of the signal is calculated by taking the mean of all the hundred samples present in the array using arithmetic average (mean) to find the average, i.e.

$$\text{Average voltage} = \frac{\text{sum of all the values in the array}}{\text{number of samples}} \times \text{voltage division} \quad \text{--------------------------} \; (9)$$

Since we took hundred (100) ADC samples per page, therefore number of samples in the equation above is 100.

Frequency:

The frequency of the analog signal was calculated by using the concept of zero crossing. We implemented three settings which were rising threshold, falling threshold or no threshold at all. The time it takes for the wave to cross zero is taken into account, where we know that

$$\text{frequency} = \frac{\text{number of samples taken before the signal crosses zero}}{\text{time before the signal crosses zero}} \text{ Hz} \quad \text{------------------------} \; (10)$$

A while loop was used to count how many samples we could take before a zero crossing.

In addition to the features, we also implemented a STROBE function so that the current display can be held stationary for a more detailed analysis of the wave. We included a SAVE function that saves the currently displayed page (i.e. the current array of 100 ADC samples) into the microcontroller's EEPROM as well as a RECALL function to display any saved page from EEPROM unto the GLCD. All these were implemented at the software level i.e. in the C code.

4.0 Construction Testing and Result

The construction and testing of the digital oscilloscope was broken-down into the different stages/units. Each of the circuit unit was first built on a bread board and tested before they were eventually built on the Vero-board. Prior to the testing on the bread board they were simulated in "Lab-center electronic proteus 7 professional" shown in Fig. 7.

First, we built the power supply unit on a standard breadboard to verify the design and connections before proceding to soldering the components on the Vero board.Thecomponentswasmounted and soldered on a Vero (strip) board and tested. After the successful test of the unit, it was housed inside a 6"x6" plastic box for protection against shock and also provided good aesthetics on the device.

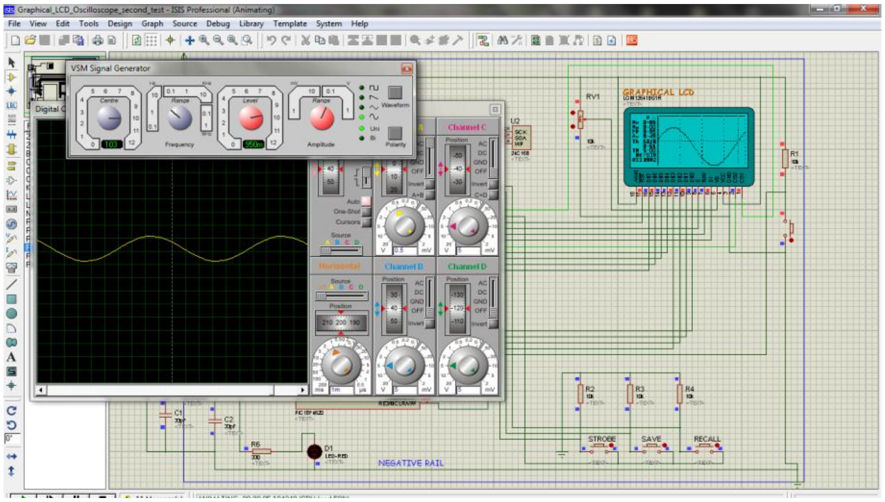

Figure 7: Proteus simulation of the digital oscilloscope when fed with a sine wave.

(a) (b) (C)

Fig.8:(a) Power supply on Vero board. (b) Housing the Oscilloscope. (C) The assembled digital oscilloscope.

4. 1 Testing

The different circuit units were tested at different stages of construction. The microcontroller code was tested on a development board (Easy PIC 6 from Mikro Elektronika®) to ensure that the code execute properly before mounting the microcontroller into the built circuit. Several software simulation of the different circuit units were run before their actual construction.

4.2 Result

The digital oscilloscope was found to meet the design requirement.

5.0 Conclusion and Recommendation

The electronic components used in the design of the digital oscilloscope were chosen based on a number of parameters; amongst which is the cheapness, ruggedness and availability of the electronic components whilst still capable of providing considerable performance when compared with the commercial ones.

A digital oscilloscope can be designed and constructed using off-the-shelf components which are cheap and readily available even though they produce considerable slower response than the professional digital oscilloscope; they can still be used for most general purpose signal analysis where the commercial ones are unavailable or too costly to be used.

Even though the constructed digital oscilloscope provides considerable performance, it still has a lot of limitations hence, a lot of room for improvements.

A major limitation of this design is its relatively low frequency limit fixed by the sampling speed of the ADC in the microcontroller, the low voltage range it can sample which is also fixed by the maximum allowable voltage that can be fed to the ADC which is fixed at 5 volts and the relatively few functions available on the oscilloscope.

One way to go will be to include additional functionalities e.g. a spectrum analyzing function, a way to analyze input signals that have different voltage range by using an auto-transformer to reduce their amplitudes, a way to sample higher frequencies etc. More input channels could also be added to enable different input signals to be analyzed at the same time and compared; also the 128 X 64 GLCD display could be replaced with a TFT screen or a big 1024 X 768 LCD screen.

Additionally, the program code can be optimizedby writing (most of) the code in assembly language to improved the frequency response hence,increasing the speed.

Finally, the whole design could be implemented using FPGAs (field programmable gate arrays) which will not only reduces noise in the circuit but also gives excellent frequency response/speed making it suitable for analyzing higher frequencies (in the MHz and even the GHz range by making use of a prescaler-MC1279).

6.0 References

[1] Tektronix, 2009.“Oscilloscope Fundamentals”, www.tektronix.com, Pg. 22.

[2] B+K Precision Corporation,2001. “Oscilloscope Applications Guidebook” Third Edition.

[3] Alan V. Oppenheim, Ronald W. Schafer,2010. “Discrete-Time Signal Processing”, Third Edition.

[4] Bruno A. Olshausen, 2000. “Aliasing”, lecture note; PSC 129 - Sensory Processes, pg. 1.

[5] Mary Anne Peters and Joseph Tylka,2010. “Digital Sampling Oscilloscope”,B. Sc. Thesis, Department of Mechanical and Aerospace Engineering Princeton University,Princeton.

[6] Wikipedia, 2013. “Oscilloscope types”, Avialable at http://en.wikipedia.org/wiki/Oscilloscope_types, accessed at 20/03/2014.

[7] Orukpe,P.E. and Edeko,F.O.2007. “Development of a Solid State Oscilloscope”, Journal of Electrical and Electronic Engineering, Vol. 2, No. 1, pg 45 – 50.

[8] George Katz,1987. “Solid State Oscilloscope”, Electronics Today International, p.25

[9] Stephan Walter,2008. “Digital oscilloscope Module with PC interface”, B. Sc. Thesis, Department of Micro engineering, Swiss Federal Institute of Technology, Lausanne