Jurnal Teknologi, 45(D) Dis. 2006: 19 – 39

PC CONTROLLED FUNCTION GENERATOR USING DIRECT DIGITAL SYNTHESIS (DDS) TECHNOLOGY FOR ELECTRICAL CAPACITANCE TOMOGRAPHY

RUZAIRI ABDUL RAHIM[1], TEO CHIN HENG[2], S. SULAIMAN[3]
& M. S. ABDUL MANAF[4]

Abstract. A function generator is widely used for experimental work and testing to generate various types of waveform (sine, square, triangle, etc) with a particular frequency bandwidth. One of its applications is in generating waveform for the ECT (Electrical Capacitance Tomography) system. ECT is used to retrieve the details of material in a flow system. An economical PC Controlled Function Generator was designed for the use of waveform generating in ECT. For the ECT system, high accuracy and frequency waveforms are needed. Three basic waveforms are required; the sine wave, square wave, and triangular wave. The objective is to build a function generator which is capable of generating high accuracy waveforms with acceptable distortion and stability. An interfacing system between a PC and the function generator was also selected and developed with the purpose of producing accurate amplitude and frequency of the waveforms. This interfacing allows remote control to the function generator for a wider usage.

Key words: Function generator, direct digital synthesis (DDS)

Abstrak. Penjana frekuensi telah digunakan secara meluas dalam pelbagai jenis ujikaji. Ia digunakan untuk menjana pelbagai bentuk gelombang (gelombang sin, gelombang segiempat dll) pada frekuensi dan lebar jalur tertentu. Satu daripada aplikasinya ialah dalam Proses Tomografi jenis Kapasitif (ECT) di mana ia digunakan untuk menyalurkan maklumat mengenai keadaan aliran. Kertas kerja ini menjelaskan bagaimana mereka bentuk sebuah Penjana Fungsi yang Dikawal oleh Komputer Peribadi yang ekonomi untuk kegunaan ujikaji makmal. Rekaan ini berbeza dengan penjana fungsi manual kerana ia hendaklah dikawal secara digital. Dalam kes ini, teknologi DDS (*Direct Digital Synthesis*) digunakan kerana cip DDS adalah digital secara semulajadinya. Analisis telah dibuat untuk memeriksa ketepatan keluaran fungsi dari segi frekuensi dan amplitudnya.

Kata kunci: Penjana frekuensi, sintesis secara terus (DDS)

1.0 INTRODUCTION

Currently, most function generators in the laboratory made use of manual control. Although software controlled function generators are available in the

[1,2,3 & 4] Control & Instrumentation Engineering Department, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor. Email: ruzairi@fke.utm.my

market, they are not widely used. This is because of their high cost. An example of a construction of a manual-controlled function generator is shown to provide a better understanding of function generators. A comparison is also made between the MAX038 waveform generator IC [1] and the AD9834 Direct Digital Synthesis (DDS) [2] waveform generator IC in order to show the advantages of the DDS as compared to other waveform generation techniques.

2.0 MANUAL FUNCTION GENERATOR

The manual function generator uses potentiometers to adjust the waveforms' frequency and amplitude. An example of a circuit of this kind of function generator built using MAX038 IC is shown in Figure 1 [1]. The voltage level, fine tuning and course tuning are adjusted by using potentiometers while the waveforms and frequency ranges are selected by using switches. Usually, the manual function generators do not provide feedback to the user regarding the condition of the output or in other word, the hardware. Therefore, the user will never know whether the required waveform is available at the output or not unless they measure it by suitable equipments.

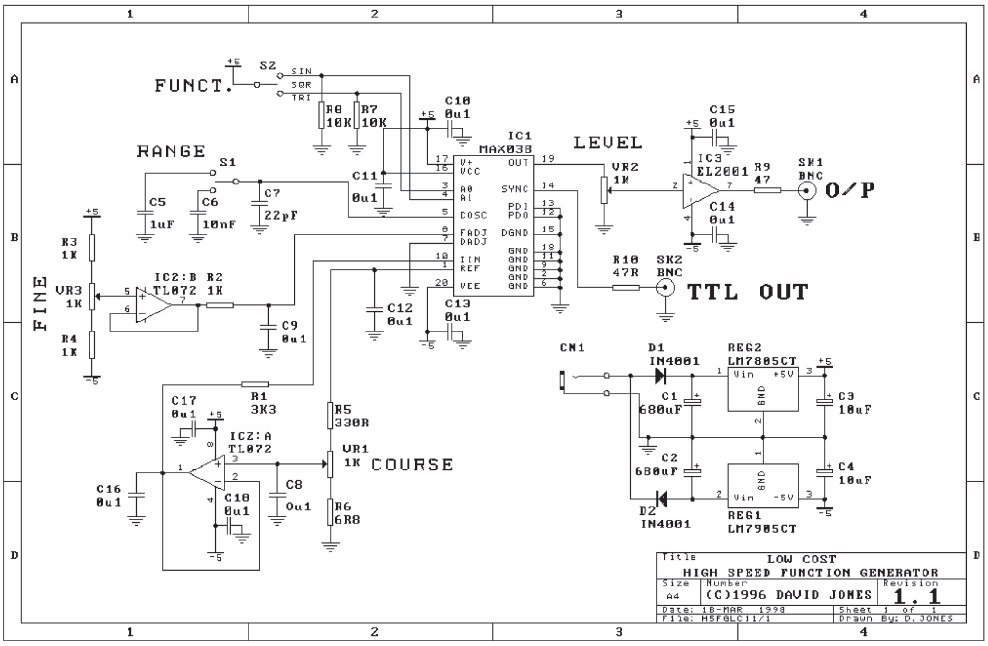


Figure 1 A schematic of function generator using the MAX038 IC

Although the manual function generator is simple and easy to use, there are several weaknesses which limits its application its usage [3]. It is often difficult for the users to adjust the frequency and amplitude of the waveforms to the exact value required because of the tuning difficulty using potentiometers. The function generator does not allow distant control of the waveform. Therefore users have to manually adjust the function generator every time they intend to change the waveform. A typical function generator does not allow interfacing and combination with other devices. However by using a PC as a main station, various devices can be connected together to form a complete testing and experimentation setup.

The above weaknesses of a manual function generator can be overcome using a PC-Controlled Function Generator.

3.0 AD9834 DDS FUNCTION GENERATOR

The AD9834 (Figure 2) provides sinusoidal, triangular, and square-wave outputs using the DDS (direct-digital-synthesis) architecture. Frequency resolution of better than 0.1 Hz can be achieved, and thus, exact coherent frequencies can be programmed. This feature is useful in digital modulation and frequency-tuning applications [4 - 6].

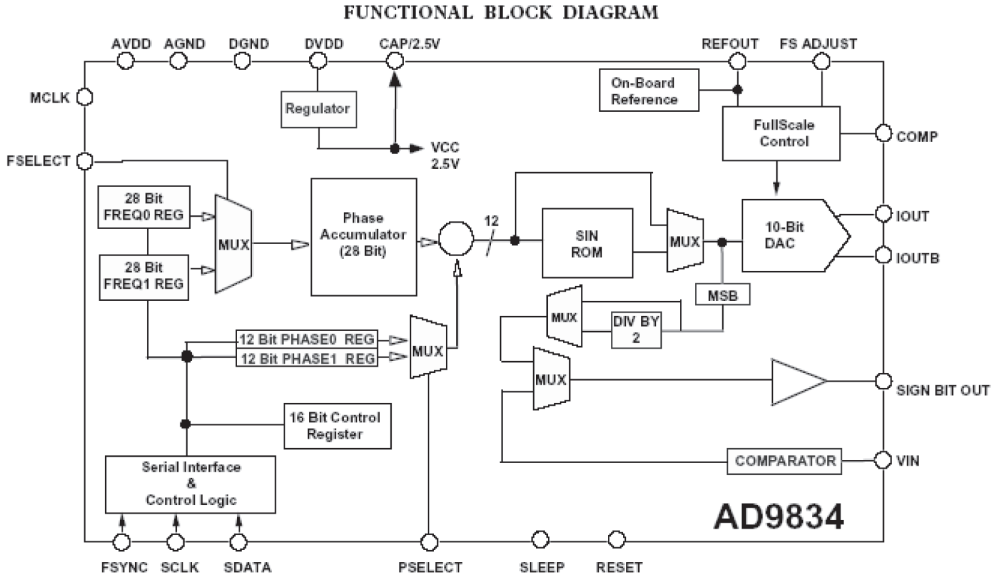


Figure 2 Functional Block Diagram of AD9834

The chip operates as an NCO (numerically controlled oscillator) using an on-chip, 28-bit phase accumulator, sine-coefficient ROM, and a 10-bit D/A converter. Typically consider sine waves in terms of their magnitude form, A(t) = sin(ωt). The amplitude is nonlinear and is, therefore, difficult to be generated. The angular information, on the other hand, is perfectly linear. That is, the phase angle rotates through a fixed angle for each unit in time. Knowing that the phase of a sine wave is linear, and, given a reference interval (clock period), the phase rotation for that period is:

$$\Delta\text{Phase} = \omega\ dt; \tag{1}$$

$$\omega = \frac{\Delta\text{Phase}}{dt} = 2\pi f\text{, and}$$

$$f = \frac{\Delta\text{Phase} \times f_{MCLK}}{2^{28}},$$

where $dt = \frac{1}{f_{MCLK}}$, and is the master clock.

Using this formula, output frequencies can be generated, knowing the phase and master-clock frequency. Theoretically, a maximum output frequency of $f_{MCLK}/2$ is possible. The DDS can be programmed by writing to the frequency registers. The analogue output from the part is then:

$$f_{OUT} = \frac{\text{MCLK}}{2^{28}} \times (\text{frequency register word}) \tag{2}$$

The outputs of the DDS have 28-bit resolution, hence effective frequency steps on the order of 0.1 Hz are possible to a maximum of approximately 20 MHz. Two phase registers are available that allow 12-bit phase resolution. These registers phase-shift the signal by:

$$\text{Phase shift} = \frac{2\pi}{4096} \times (\text{phase register word}) \tag{3}$$

The AD9834 contains 2 frequency registers and 2 phase registers. Each frequency register has a size of 28 bits while each phase register has 12 bits. In this project, only the frequency register 0 and phase register 0 are used. The analog output from the AD9834 is:

$$f = \frac{\Delta Phase \times f_{MCLK}}{2^{28}} \tag{4}$$

where ΔPhase is the value contained in the selected frequency register. Therefore, the content of the selected frequency register will be $\frac{f \times 2^{28}}{f_{MCLK}}$.

The signal will be phase shifted by:

$$\frac{2\pi}{4096} \times PHASEREG \tag{5}$$

where PHASEREG is the value contained in the selected phase register.

When writing to a frequency register, bits D15 and D14 give the address of the frequency register. Since only frequency register 0 is used, D15 and D14 are set to 0 and 1, simultaneously. When writing to the frequency register, two consecutive writes to the same address are performed, as the frequency register is 28 bits wide. The first write contains the 14 LSBs while the second write contains the 14 MSBs. For this mode of operation, the control bit B28 (D13) is set to 1. After changing the content of frequency register, the output will change accordingly after 7/8 MCLK cycles.

Summary of Features in AD9834 [2] are as follows:

(i) The AD9834 provides a high resolution frequency for three different types of waveform: sine wave, square wave and triangular wave. The frequency selection for the chip is done by altering the content of the frequency register in the AD9834. The chip is digital in nature, allowing easier communication with the PC.

(ii) Since the AD9834 synthesizer has a crystal locked output frequency, there is no temperature and time drift for the frequency. This provides a very stable output frequency.

(iii) Theoretically, the AD9834 is able to generate a maximum output frequency of $\frac{f_{MCLK}}{2}$. The maximum master clock that can be used by the AD9834 is 50 MHz. Therefore, the chip is able to generate a highest frequency of 25 MHz. However, a 50 MHz master clock will result in a more difficult circuit construction since it is considered as a very high frequency range.

(iv) AD9834 from Analog Devices is only available in the Thin Shrink Small Outline Package (TSSOP) package. This is an extremely small surface mount package. A PCB has to be built when constructing the circuit for the chip.

4.0 SYSTEM DESIGN

There are three basic subsystems in the PC Controlled Function Generator as shown in Figure 3.

(a) Software
Visual Basic 6.0 is used to create a Graphical User Interface (GUI) to allow the user to select the type, amplitude and frequency of the waveform that they want to generate. After that, the GUI is able to send the required data to the function generator through an interfacing system. Visual Basic 6.0 is chosen as the programming platform because of its simple application. Visual Basic 6.0 is rather slower as compared to other programming platforms such as Visual C++. However this does not cause much effect to the system because the data transmission does not require a fast speed [7].

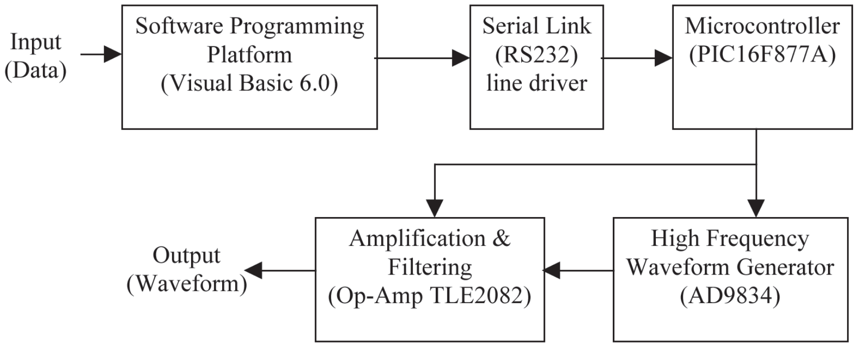

Input (Data)
Software Programming Platform (Visual Basic 6.0)
Serial Link (RS232) line driver
Microcontroller (PIC16F877A)
Amplification & Filtering (Op-Amp TLE2082)
High Frequency Waveform Generator (AD9834)
Output (Waveform)

Figure 3 Block diagram

(b) Interfacing
A serial link is used to send the data from PC to the Function Generator. There are only three types of data to be sent to the Function Generator which are the type of waveform to be generated, its amplitude, and its frequency.

(c) Hardware
The data received from the PC is input to a microcontroller through a serial link. This microcontroller will convert the data to suit the requirement of the Function Generator in order to generate the waveform. The modified data is then sent to the Function Generator IC through output pins.
Function Generator IC is already available in the market; therefore it is used as the main part to generate the required waveforms. This IC generates the type

and frequency of the waveform as required. However, the amplitude of the output waveform is constant in small value depends on the manufacturer specitications.

The output waveform from the IC is then amplified to required amplitude based on the data from the PC. To obtain a good quality waveform, that is low distortion and stable, the waveform is then filtered using low pass filter to eliminate high frequency noises.

The input data includes the required type, frequency and amplitude of a waveform. It is keyed in to the computer through a GUI built by using Visual Basic 6.0. After that the data is sent to a microcontroller through a serial link. The microcontroller then communicates with the AD9834 and TLE2082 in order to provide an accurate waveform at the output. The block diagram is shown in Figure 3.

4.1 Software Design

Microsoft Visual Basic 6.0 was chosen to build the GUI for the project. The data sent to the hardware does not require a high speed. It is a one-way data sent. Visual Basic 6.0 is more than enough to fulfill the requirement. The platform has a built-in Microsoft Communication Port control allowing easy settings of serial communication port on a PC [8]. Figures 4 to 8 show the flowchart for the GUI programming.

4.1.1 Main Program Loop

When the GUI is first open, the settings for the waveform will be initialized (Figure 4). After that, each time the Reset button is pressed, the initialization of the GUI runs again.

After the initialization, the GUI will wait for the Generate command from the user. If the user wishes to change any setting to the type of waveform, amplitude or frequency, it may be done at anytime. After the user has decided on the required type of waveform, amplitude and frequency, he/she may press the Generate command button to tell the function generator to generate the output waveform which meets the settings on the GUI.

The user may only select three types of waveform on the GUI: sine wave, square wave and triangular wave. There are also only three selections of amplitude available, which are 1V, 4V and 9V. For the frequency setting, the user may key in any frequency in the range from 1 Hz to 1 MHz in a 0.01 Hz resolution.

For every change the user make for the type of waveform and amplitude, the GUI will draw a preview of the waveform that is to be generated on the user's command. The preview is drawn on a graph template on the GUI, showing the type of waveform and its amplitude. Due to the many variety of frequency that can be selected, the preview waveform does not have the information on it.

If the user wishes to terminate the GUI, the Exit button may be pressed and the program will end (Figure 5).

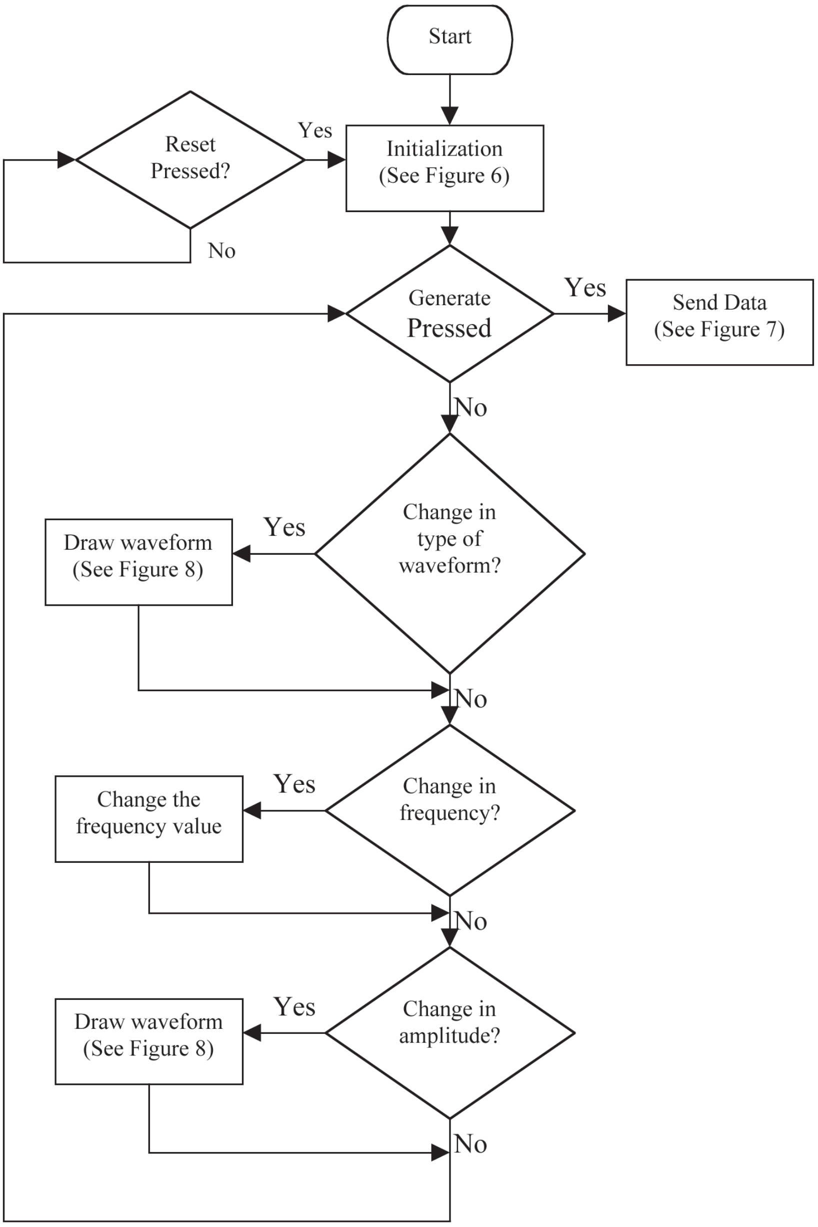

Start
Reset Pressed?
Yes
No
Initialization (See Figure 6)
Generate Pressed
Yes
Send Data (See Figure 7)
No
Change in type of waveform?
Yes
Draw waveform (See Figure 8)
No
Change in frequency?
Yes
Change the frequency value
No
Change in amplitude?
Yes
Draw waveform (See Figure 8)
No

Figure 4 Flowchart of main program for the GUI including the reset function

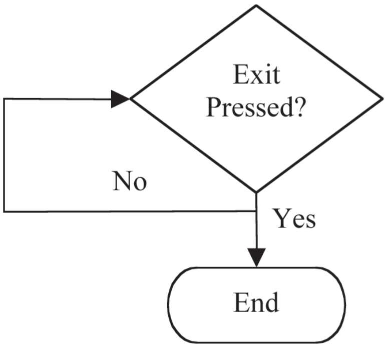


Figure 5 Flowchart for the Exit function

4.1.2 Program Initialization

Once the GUI is started, initialization will take place. The GUI will internally set the type of waveform as a sine wave, with frequency of 1000 Hz and amplitude of 4V (Figure 6). A preview to the initialized waveform is drawn to inform the user about the presets of the GUI. After the settings, all the data will be sent to the hardware to generate the initialized waveform (Figure 7).

4.1.3 Sending Data through Communication Port (Serial Link)

Before sending all the information to the microcontroller, the GUI will firstly convert the data into a form to allow manipulation of data by the microcontroller

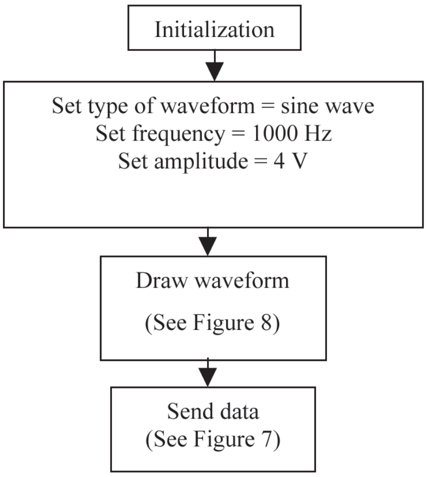


Figure 6 Flowchart for initialization of the GUI

later. The variables to be sent to the microcontroller are the amplitude which contains the amplitude information, waveform which contains the type of waveform and the 4 byte which contains the information of the frequency.

First, the GUI will detect the amplitude selected by the user. There are three amplitude options for selection. If the amplitude is 1V, the amplitude variable will be set to 1. The amplitude will be set to 2 and 4 according to the corresponding amplitude setting, which are 4V and 9V. These values, when converted into ASCII code are 31, 32 and 34. Therefore, the microcontroller will only have to detect bit 0, bit 1 and bit 2 of the variable in order to get the information of the selected amplitude.

The same algorithm is used to represent the waveform information. The waveform variable will be set to 1, 2 and 4 to represent the sine wave, square wave and triangular wave respectively.

After that, the frequency setting will be converted into the value to be entered into the AD9843's frequency register based on Equation 1. Therefore:

$$\Delta Phase = \frac{Frequency\_setting \times 2^{28}}{f_{MCLK}} \tag{6}$$

where

ΔPhase is the value to be inserted into the frequency register of AD9834,
f_{MCLK} is the master clock provided to the AD9834 (which is 20 MHz in this project).

After calculating the ΔPhase, it is then converted into hexadecimal value because the microcontroller will only recognize hexadecimal value. The maximum hexadecimal value will not exceed 4 bytes (32 bits). This is because the maximum setting of frequency is 1 MHz. After ΔPhase has been calculated and converted into hexadecimal value, it will become CCCCCD$_{16}$. The hexadecimal value will be separated into 4 bytes of frequency data, namely byte1, byte2, byte3 and byte4. This is because the RS232 system will send only 8 bits (1 byte) of data at one time.

After the data conversion is completed, the MSComm Port is opened and the data is sent to the serial link in the following sequence: waveform, amplitude, byte1, byte2, byte3, byte4. After that, the MSComm Port is closed again until the next data sending.

4.1.4 Waveform Preview

Every time the amplitude or the type of waveform is changed, the GUI will draw a waveform representing the preview of the waveform to be generated if

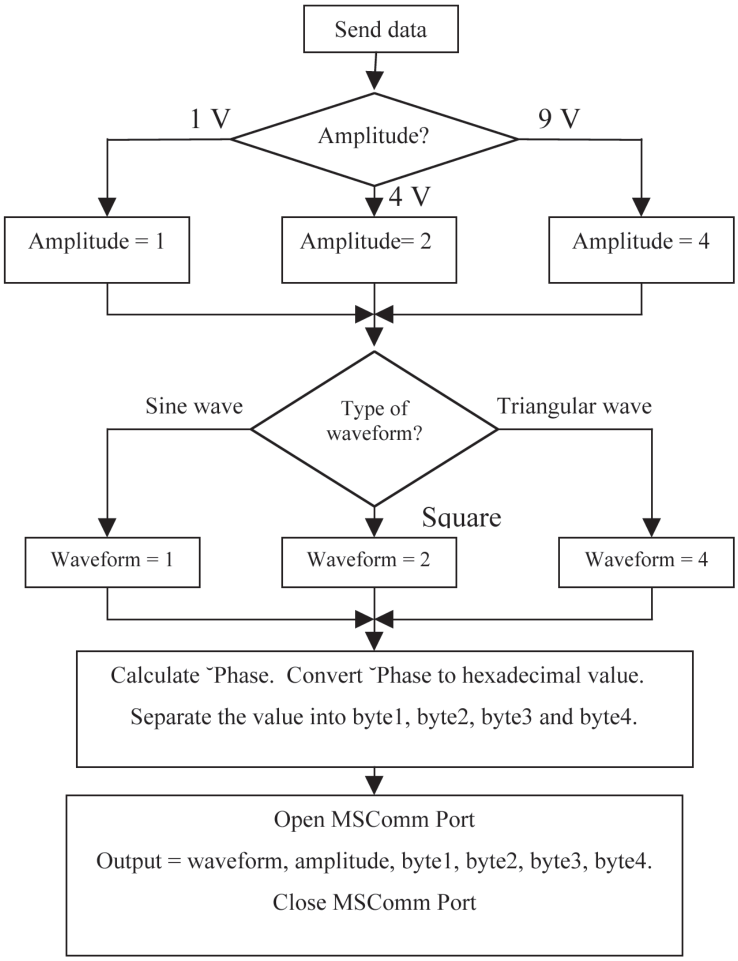


Figure 7 Flowchart for sending the data

the user pressed the Generate command button. The flow chart of drawing a waveform is shown in Figure 8.

4.2 Serial Link (RS232)Line Driver

A serial link was chosen as the communication between the computer and the microcontroller in this project. A serial link was chosen instead of a parallel link, because serial link has many advantages. A serial cable can be longer than a parallel cable [3]. The serial port transmits a ‘1’ as -3 to -25 volts and a ‘0’ as +3 to +25 volts where as a parallel port transmits a ‘0’ as 0V and a ‘1’ as 5V. Therefore the serial port can have a maximum swing of 50V compared to the

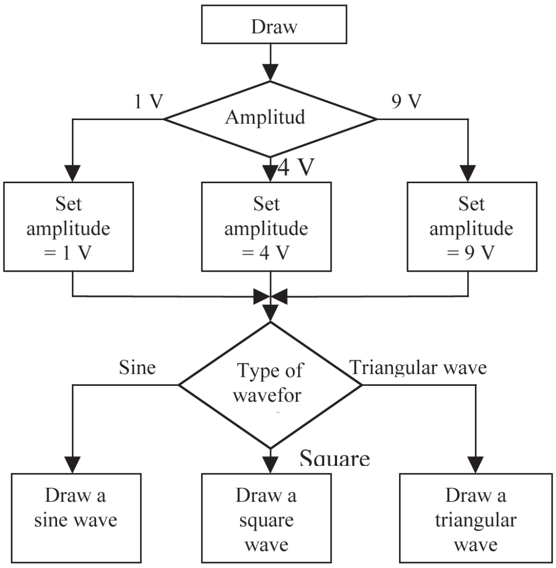


Figure 8 Flowchart for drawing waveforms

parallel port which has a maximum swing of only 5 Volts. Less wires are needed in serial communication. This will reduce the cost of installation. The microcontroller is chosen in this project is of type PIC16F877A which has a built - in SCI (Serial Communications Interfaces). Serial communication reduces the pin count of this MPU. Only two pins are commonly used, Transmit Data (TX) and Receive Data (RX) compared with at least 8 pins if the 8-bit parallel method is used.

4.3 Microcontroller PIC16F877A

A microcontroller is used in order to receive data from the computer using the USART system, and convert the data accordingly to be sent to the AD9834 and the amplifier circuit. A microcontroller is chosen instead of a microprocessor, because microcontroller have many built in peripherals, such as timers, PWM modules, analog to digital (ADC), Synchronous Serial Port (SSP) with SPI (Master Mode) and I2C (Master/Slave), Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) and Parallel Slave Port (PSP). Besides, it supports 368 RAM with the speed of 20MHz, which processes data much faster than a microprocessor.

The microcontroller is programmed so that it is able to initialize the AD9834 and amplifier circuit, receive the 6 byte of serial data from the serial link, store the data into 6 variables, convert the information in a way that is suitable to be sent to the hardware, send the information of type of waveform and frequency to the AD9834, and send the information about the amplitude to the amplifier circuit.

4.4 Hardware

4.4.1 Amplification

Op-amp is a common method to amplify a signal. However, there might be some problem to control it digitally. For amplitude selection using op-amp, analogue multiplexers need to be used. A basic circuit for an amplifier using the TLE2082 op-amp is shown in Figure 9 [9].

The circuit requires a ± 15 volts supply to provide a maximum voltage swing of 30 volts peak to peak. It is an inverting amplifier where R*f* and R*g* determines the gain base on the formula:

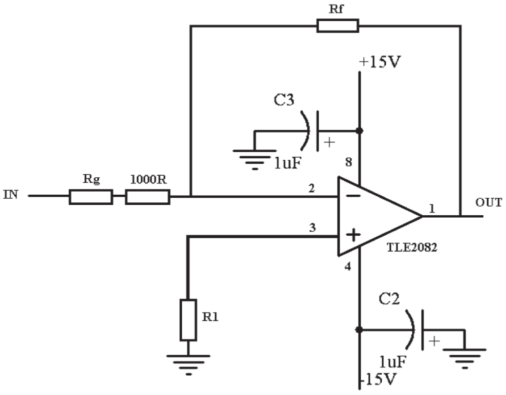


Figure 9 Schematic for basic amplifier using TLE2082

$$\text{Gain} = \frac{Rf}{Rg + 1000} \tag{7}$$

In this project, *Rf* is fixed at 20 kΩ while *Rg* is determined by the user's required amplitude through selection by an analogue multiplexer (MC14052), as shown in Figure 10. Therefore outputs A and B from PIC will select the suitable *Rg* in giving the required amplitude. This is shown in Table 1.

The output sine wave and triangular wave from AD9834 have almost constant amplitude of approximately 0.55 volts. Therefore, the larger the gain, the bigger

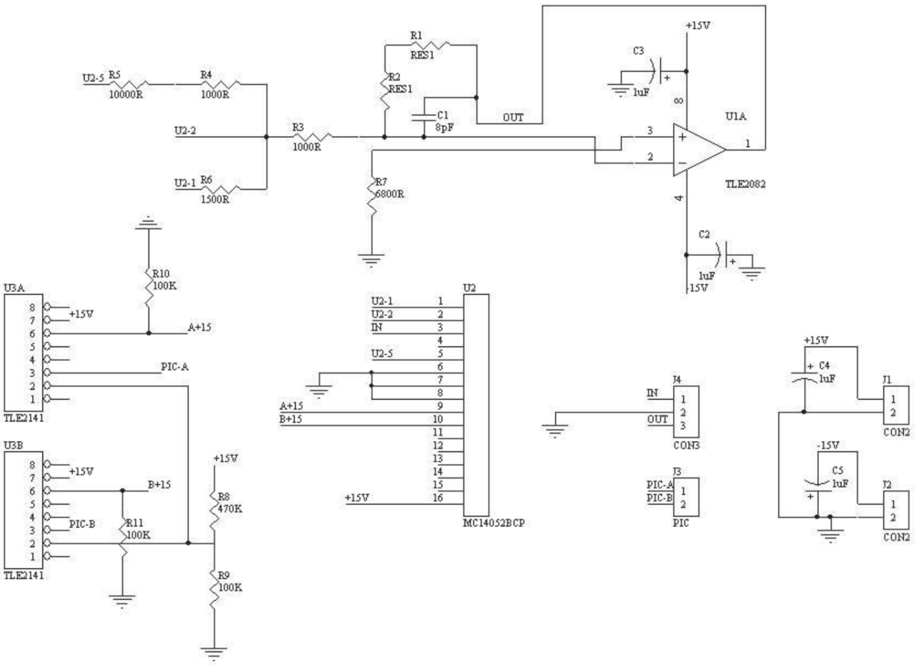


Figure 10 Schematic for PC Controlled Amplifier along with filter

Table 1 R_g and Gain for every combination of A and B

A	B	Rg	Gain
0	0	1.5 kΩ	8
0	1	200 Ω	16.6667
1	0	11 kΩ	1.6667
1	1	Open circuit	Open circuit

the amplitude will be. The output from the amplifier will have amplitude of 0.55 × Gain, thus enabling amplitude selections of 1V, 4V and 9V.

Resistor R1 is used to eliminate the DC offset that is caused by the AD9834 in order to provide a purely AC output. The value of R1 is hence 6.8 kΩ.

4.4.2 Filter

The filter is required to eliminate high frequency noises. Figure 11 shows how a capacitor, C1 is added into the amplifier circuit to enable the filtering effect on the circuit. Since the scope of the project only requires highest frequency of

100 kHz, any signal more than 1MHz is decided to be filtered out. The bandwidth of the circuit is from 0Hz to $\frac{1}{2\pi RfC1}$ Hz. Therefore, $C1 = \frac{1}{2\pi Rf \times 10^{6}} \approx 8pF$. A 8pF capacitor is installed at the location of C1 to provide filter function.

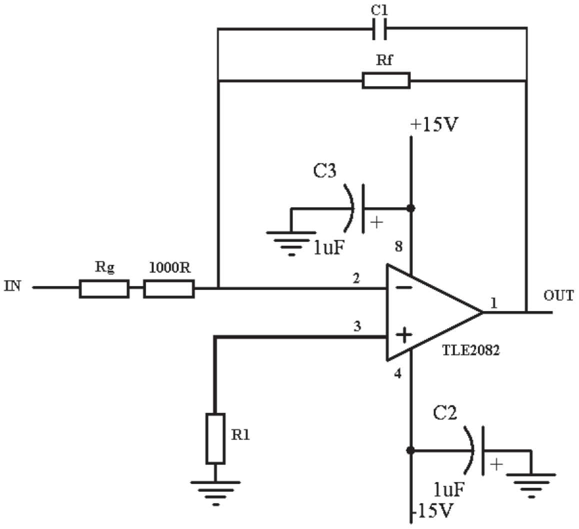


Figure 11 Schematic for basic amplifier and filter using TLE2082

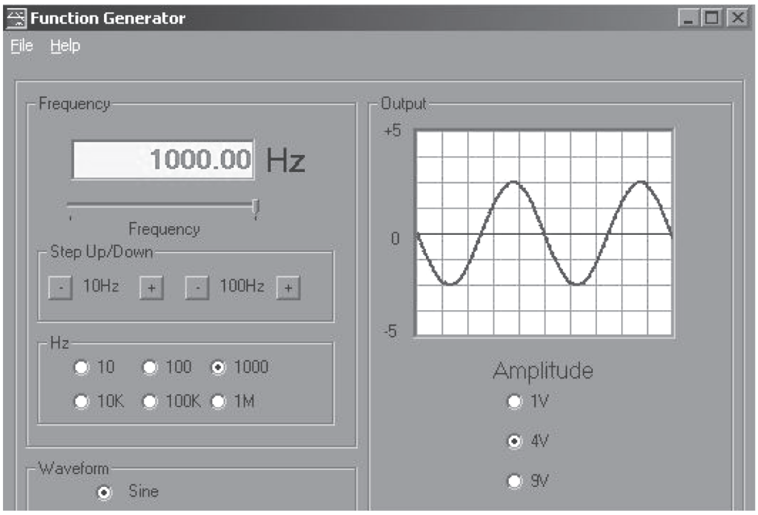
5.0 RESULTS AND ANALYSIS

The GUI was successfully built by using the Visual Basic 6.0 software programming platform. It provides an easy navigation between the user and the function generator in selecting the desired type of waveform frequency, and amplitude.

The PIC16F877A, along with the serial link, was constructed. It receives data from the GUI and send to the amplifier circuit and AD9834 after suitable conversion. The AD9834 was constructed on a PCB to enable higher quality of waveform generation. In the mean time, the PC Controlled Amplifier and filter was constructed on a strip board. The hardware works according to requirements. The following sections will explain the results for some parts of the system and analysis on the waveform.

5.1 The GUI

Figure 12 shows the screen capture of the GUI. The navigation is basically divided into four sections: type of waveform, frequency selection, amplitude selection and command buttons. The output of the waveform can be previewed from the graph above the amplitude selection section.



Frequency

1000.00 Hz

Frequency

Hz

○ 10 ○ 100 ● 1000

○ 10K ○ 100K ○ 1M

Waveform

● Sine

Amplitude

○ 1V

● 4V

○ 9V

Figure 12 The Graphical User Interface

The types of waveform available for the user to select are sine wave, square wave and triangular wave. The amplitude available are only 1V, 4V and 9V. These are limited by the capability of the function generator.

As for frequency selection section, the option buttons 10Hz, 100Hz, 1000Hz, 10KHz, 100KHz and 1MHz set the upper limit of the output frequency. These options are very useful when the user does not want the output frequency to exceed a typical range during experiment or other situation. These options also allow easier design in hardware when more amplitude selection and compensation for different frequency range is required. To determine the frequency that is to be generated, the user can directly type in the required frequency into the text box using a keyboard. In other cases, the user can also choose to use a mouse to navigate with the frequency selection section through the sliding bar and the step up/ step down buttons. These many ways of frequency selections makes the GUI more user friendly and easy to use.

5.2 Waveform Analysis for AD9834 Output

Analysis had to be done for the waveform generated by AD9834 before amplification. This is to gather information for the amplifier design. Examples of sine wave, square wave and triangular wave of 1 kHz are shown in Figure 13.

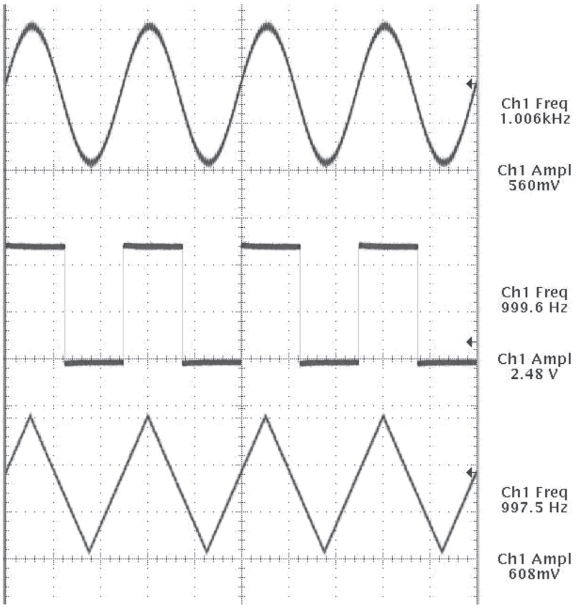


Figure 13 Examples of waveforms generated by AD9834

The required analysis was to determine the amplitude for each waveform generated from the AD9834 IC. A set of amplitude data was measured and recorded over the frequency range and a graph was plotted to show the result clearly. All measurements were made using the Tektronix TDS 3012 Two Channel Color Digital Phosphor Oscilloscope.

Figure 14 shows the relationship between the amplitude of the sine wave from AD9834 over the frequency range.

From the graph, the amplitude for the waveform is constant at 0.55 volts before 1MHz. Only the amplitude lower than 1MHz is concerned because the project does not require a high frequency of waveform. The constant value of 0.55 volts is therefore taken as the representative amplitude for waveform input to the amplifier circuit.

Figure 15 shows the relationship between the amplitude of the square wave from AD9834 over the frequency range.

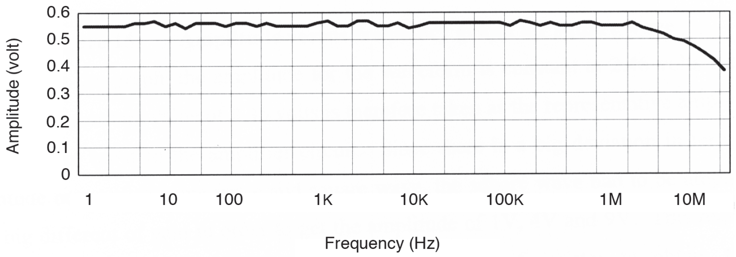


Figure 14 Amplitude versus frequency graph for sine wave

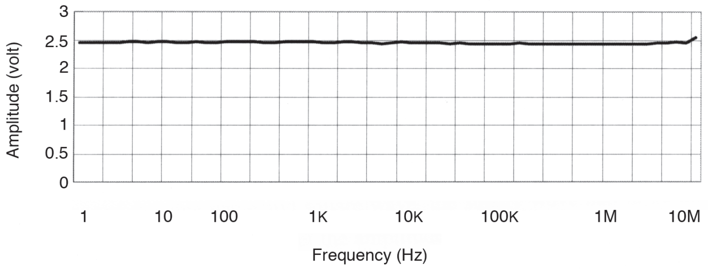


Figure 15 Amplitude versus frequency graph for square wave

From the graph, the amplitude for the waveform is constant at 2.46 volts before 1MHz. The constant value of 2.46 volts is therefore taken as the representative amplitude for waveform input to the amplifier circuit. Since there is a big deviation in terms of amplitude between the sine wave and square wave, the square wave had to be amplified with big different of gain in order to get the amplitude of 1V, 4V and 9V. This requires another analogue multiplexer to select different pairs of resistor to obtain those amplitudes. In this project, only one analogue multiplexer (MC14052) is available for utilization. Therefore, the square wave is not amplified and the 2.46 amplitude is sent directly to the output.

Figure 16 shows the relationship between the amplitude of the triangular wave from AD9834 over the frequency range. From the graph, the amplitude for the waveform is constant at 0.61 volts before 1MHz. The constant value of 0.61 volts is therefore taken as the representative amplitude for waveform input to the amplifier circuit. Since this constant value is close to 0.55 volts, which is the constant amplitude for the sine wave, the gains are designed to be shared by both sine wave and triangular wave. In this way, only one analogue multiplexer is used.

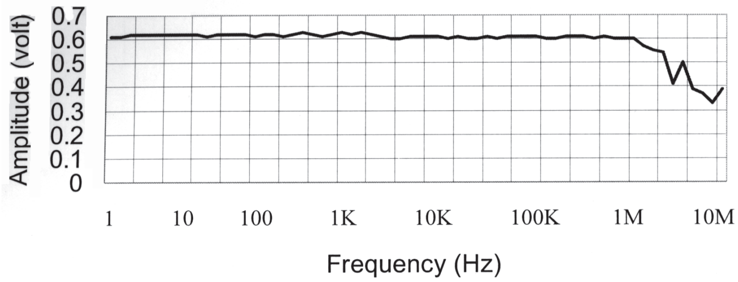


Figure 16 Amplitude versus frequency graph for triangular wave

5.3 Analysis on the Accuracy of Output Frequency

One of the most important aspects in the scope of the project is the accuracy of the frequency. The output waveform should have a frequency as close as possible to the user desired frequency. Therefore, output frequency over the frequency setting was analyzed. Six measurements were made to the frequency for some selected frequency setting for each type of waveform. An error percentage was computed based on the largest deviation of output frequency to the frequency setting. This was done by using the formula:

$$\text{Error} = \frac{\text{Max}(|\text{frequency setting} - \text{measured frequency}|)}{\text{frequency setting}} \times 100\,\% \qquad (8)$$

where i = 1, 2, 3, 4, 5, and 6.

The results are then plotted into graphs as shown in Figures 17,18 and 19.

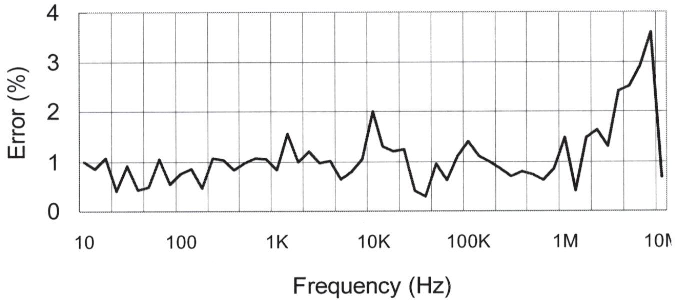


Figure 17 Percentage of error versus frequency for sine wave

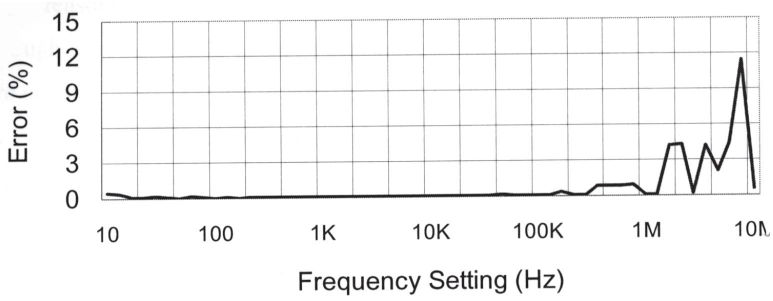


Figure 18 Percentage of error versus frequency for square wave

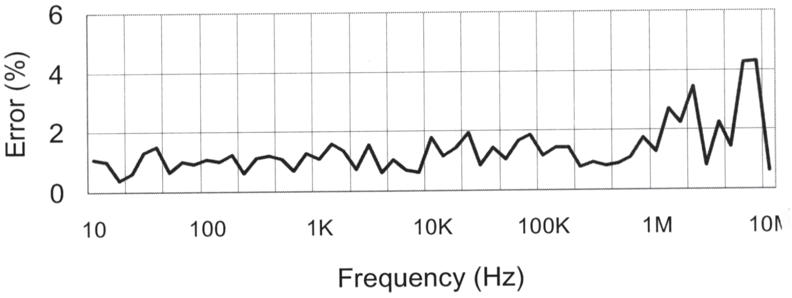


Figure 19 Percentage of error versus frequency for triangular wave

In this project, a frequency error that is lower than 5% is acceptable. For the sine wave, the percentage of error remains lower than 4% at any frequency setting, even for frequency up to 10 MHz. This result shows that the error of frequency for sine wave falls under the acceptable range. For frequency setting equals or below 1 MHz, the result is even better which consists of frequency error lower than 2%.

For the square wave, there is a high shot of frequency error at frequency setting of 9 MHz. This error reaches 11%. However, for frequency setting below 1 MHz, which is more important in this project, it is even lower than the result shown by the sine wave. The error at this frequency range is not less 1%.

The graph for the triangular wave shows an almost the same result as the sine wave. Although the error almost reaches 5% at frequencies above 9 MHz, the error is still lower than 2% at frequencies below 1 MHz.

The reasons for the existence of errors in the output frequency are as follow:

(i) Although a stable crystal clock of 20 MHz is used as the master clock for AD9834, there are still some changes in its value which differ the clock value. This deviation may reach 0.01 MHz. Since all the calculation for input to frequency register was calculated by assuming a perfectly accurate master clock, the output frequency will change in small value along with the change in master clock, if available.

(ii) The calculation of value to be sent to the frequency register is based on equation 1. The result might not be an integer. However, it is then converted to an integer because the frequency register in AD9834 can only accept integer. The small deviation while rounding up the value will cause a slightly inaccurate output frequency. During the designing stage of the system, this error was already expected to be acceptable.

(iii) Since the readings of output frequency were taken directly by observing the automatic frequency calculator in the Tektronix TDX 3012 digital oscilloscope, its accuracy depends on the performance of the scope. Due to the environmental disturbance and other relevant factors, the waveform entering the oscilloscope is not fully stable. This causes the frequency readings to change according to the unstable waveform. However, this factor contributes to a very small frequency error and therefore may be neglected.

6.0 CONCLUSIONS

A function generator which is capable to generate sine wave, square wave and triangular wave with accurate frequency and amplitude had been built. The frequency selection can be made in the range from 1 Hz to 1 MHz in smallest resolution of 0.01 Hz. In this frequency range, the amplitude for the output waveform is guaranteed to be accurate only for the range from 10 Hz to 100 KHz, as stated in the scopes of the project.

The available amplitude for sine wave and triangular wave are 1 volt, 4 volts and 9 volts. These amplitudes are suitable to meet many experimental requirements. However, the square wave is only provided with one selection of amplitude, which is 2.5 volts.

REFERENCES

[1] IC MAX038 *pdfserv.maxim-ic.com/en/ds/* MAX038.*pdf* Accessed December 2004

[2] AD9834 - Low Power *www.analog.com/en/prod/*0,,788_843_AD9834,00.*html* Acessed January 2005

[3] Dayton R.L, 1988. Guide to Integrating Digital Services: T1, DDS, and Voice Integrated Network Architecture. New York: McGraw-Hill.

[4] Direct Digital Synthesis, *www.altera.com/products/ip/altera/t-alt-dds.html* [5] Direct Digital Frequency Synthesis *www.hit.bme.hu/~papay/sci/DDS/start.htm.* Accessed January 2005

[6] Schewel, J 1999. Direct digital synthesis: some options for FPGA implementation (Paper #: 3844-02) SPIE Proceedings Vol. 3844 ISBN: 0-8194-3437-X, 180 pages Published 1999 Meeting Date: 09/19 - 09/22/99, Boston, MA, USA.

[7] Zak, D. 1999. *Programming with Microsoft Visual Basic 6.0, Course Technology*, 1999. Preston USA

[8] Schneider. D I. 1999. *An Introduction to Programming using Visual Basic* 6.0. Upper Saddle River, N.J: Prentice-Hall.

[9] Traister R J. 1989. *Operational Amplifier Circuit Manual.* San Diego, CA: Academic Press.