Al-Khwarizmi Engineering Journal, Vol.4 , No.1 , pp 48- 57 (2008)

Al-khwarizmi Engineering Journal

Microcontroller-Based Function Generator

Riyadh A.H. AL-Helali
Electrical Engineering
AL-Mustansiriyah University

Dr. Ismail A. Mohammed
Electrical Engineering
Baghdad University

Ammar I. Abdullah
Computer Electrical
Ministry of Work & Social Insurance

(Received **29** May 2007 ; accepted 5 February 2008)

Abstract:

This paper describes a microcontroller-based function generator system. By the function generator sine wave, square wave, quasi-square wave, saw-tooth and triangular waveforms are generated over a wide frequency range according to user requirements. By utilizing processing capabilities of the microcontroller the hardware is minimized exceedingly. The output waveform shapes are digitally-controlled to achieve the required wave shape. The single chip microcomputer of waveform generation equipment offers the possibility of improvements in manufacture reliability, maintenance and servicing and increased control flexibility. The system is built and tested. The results of test were satisfactory and appreciated by test engineers at different centers of ministry of communications.

Keywords: microcontrollers & microcomputers, programmable controllers, waveform generation.

Introduction:

Educational institutions and colleges as well as test engineers need a function generator that generates certain shapes of waveforms at a wide range of frequency. The potential and versatility of function generation techniques have not yet been fully realized, and in the past have been limited by the conventional hardwired an analogue and digital circuits which were available [1,2]. This complicates the circuit and increases the cost of the instrument.

The operational advantage of the ML2036 (Micro Linear) [1] is only a sine wave generator capable of producing a sine wave signal with a frequency from DC to 50 kHz and it only needs a few components but has limited frequency range, while the experimental circuit of the HSP45102 (programmable generator) [1] used the centronic experimental board. The control software [1] is written in Turbo Pascal 6, thus, the experimental circuit operation depends on the PC, which passes through languages translators via compilation process and assembly process.

The control program that is fitted into the on-chip ROM of the microcontroller-based system deals with interrupts to generate the required function [2]. Since the important feature of microcontrollers is the built-in interrupt system.

A microcontroller-based system is numerically attractive since it offers increased flexibility, particularly in choosing the required frequency among a wide range of frequencies. These advantages inevitably result from a reduction in the complex control circuitry, which may be progressively replaced by microcontroller software. Then It is possible to generate any frequency waveform is previously assigned without altering the hardware.

To reduce the components and instrument cost a software-based system is to be designed and built. It is possible to proceed directly and attempt to implement, design a microprocessor, one of the previously developed hardware analogue or discrete digital-control technique [3]. However, to take a full advantage of the shift in emphasis from hardware to software design, and make effective use of the computing power of the microprocessor, a new and more fundamental approach to system design is required.

By building algorithms to generate the various waveforms and control programs the required waveform can be generated with a minimum number of components. Reducing the components and adopting software increase the reliability of the system and flexibility of the system operation based technique [3,4].

Real-time generation of digital waves

The “real-time” generation of a digital wave using an 89C51-version microcontroller utilizes an algorithm that consists of many subroutines to produce the required waveform. Therefore, each wave takes subroutine execution time which is different from that of another wave generation and depending on the usage of the instruction sets to form the right creation with exact time interval.

The approach to be used for generating a periodic quasi-square wave on a certain output port of the microcontroller with the required frequency is to create intervals of various lengths for setting the output port in one hand and resetting it in the other hand. It is possible to create a quasi-square wave with variable duty cycles, thus, the software running the timer is the best choice to create time delays.

The creation of a square wave used the same approach of a quasi-square, with a duty cycle of 50%. Another approach for creating it by using time interrupts; with time interrupts enabled [4,5], the event that generates the interrupt is the setting of the timer flag upon overflow of the timer registers. The triangular wave can be generated using a software up/down counter, the rate at which this counter is incremented (or decremented) determines the frequency and accuracy of the generation process. It also can be achieved based upon either resetting the up/down counter at the start of each cycle or deriving the interrupt signal to the CPU from the up/down timers [3].

The saw-tooth wave algorithm is achieved by using software based on up/down counter that is incremented with a suitable STEP value for resetting the maximum values register to produce the required frequency.

The “real-time” generation of a sine wave using a microcomputer algorithm is time consuming [3,4]. For example, a Zilog Z80 microprocessor with a 2.5 MHz clock takes about 800 μs to calculate one sine value [3]. It is therefore inappropriate in high frequency applications to calculate the sine wave values, as they are required in “real-time”. An alternative approach is to store the sine values in a lookup table, which is preprogrammed into on-chip code memory (ROM). Alternatively, the sine values may be stored in external volatile memory (RAM), the values having first been calculated by the microprocessor during the initialization period prior to generator startup.

The memory requirement, efficiency of operation and accuracy of output waveform depends on the number of sample values defining a cycle of the sine wave and their resolution. If, for example, the values are taken at 0.5^0 intervals, then a complete sine wave cycle is defined by 720 values. The sine wave could be defined at a greater number of sample points, but the memory requirement is proportionally increased.

The resolution of each lookup table value must be specified on the basis of the type of generation algorithm to be performed and the required accuracy of the width output values. For most applications, an 8-bits word length is sufficient and this is particularly appropriate if an 8-bit microprocessor used.

The memory requirement is reduced if only one half or even a quarter of a complete sine cycle is stored in a lookup table [2]. In either case, the bit definition is increased by one since it is no longer necessary to indicate the sine of each sample value stored.

The accessing of values from the lookup table may most easily be indicated using a pointer. If a complete generated sine wave cycle is stored in memory, then the pointer cycles the lookup table. A similar procedure may be followed if only half a cycle of the sine wave is stored [3]; although, in this case, it is necessary to include a flag to indicate whether the case value obtained is of positive or negative sign. This flag is also required if only a quarter of a cycle is stored. In this case, the pointer moves up and down the lookup table instead of in one direction.

Hardware-based system

In This system, the microcontroller is only required to generate the variable-frequency digital waveforms. A typical system arrangement is illustrated in Figure 1. The system consists of the following blocks**:**

1. 89C51 Microcontroller.
2. Hexadecimal Keypad.
3. Intelligent dot matrix alphanumeric display [6].
4. Digital–to–Analogue Converter (DAC) [2,3,4].
5. Buffer stages to isolate the outputs.
6. Amplifier Stage.

There are two modes of operation, which take two stages, and generate two groups of waveforms; group 1 includes generation of triangular, saw-tooth and sine waveforms, while square and quasi-square waveforms belong to group 2.

Interfaces to keypads are common for microcontroller-based designs, the keypad contains 16 keys, which are connected to interrupt control pins of the microcontroller. During each interrupt, the memory is accessed and takes variable time for each generated waveform.

While the 2416 [6], is a four digit, 5*7 dot matrix intelligent alphanumeric LED display modules complete with built-in CMOS Driving circuitry and it is connected to port 0 of the 89C51. Each character in any memory location can be addressed independently and will continue to display the character last written until another one replaces it.

Interfacing the real world often requires generating or sensing analog conditions this design used two transistors, two capacitors, a potentiometer, an LM301 op amp. [5], which represents the amplifier stage and MC1408L8 [4], is an 8-bit digital-to-analog converter (DAC). The eight data inputs to the DAC are driven from port 1 on the 89C51.

Writing different values on port 1 and adjusting the 1K potentiometer, the output should vary from 0 volts (p1 = 00H) to about 10 volts (p1 = FFH).

The input command information, which relates the wave shape and its required frequency, is entered through the keypad. By pressing the appropriate key, an interrupt signal is driven that executes the interrupt service routine (ISR) algorithm, then the memory is accessed and output the required waveform. Each key of the keypad corresponds to a certain (ISR). There are (12) ISRs carrying out the following functions**:**

(i) Defining the frequency.
(ii) Adjusting the display digits using (4) keys.
(iii) Selecting the right frequency page of the program using (2) keys.
(iv) Choosing the wave shape using one of (5) keys.

The linear-mode operation that generates waveforms of group 1 through port 1 of the microcontroller to the digital-to-analogue converter then produces the analogue signal at low power level. The levels of voltage and current can be amplified by using an amplifier stage to get the required specifications.

The switching-mode generation that generates waveforms of group 2 through one pin of port 4 of the microcontroller and it is also amplified using the same amplifier stage.

An optocoupler is used as a buffer stage between the microcontroller and the power circuit [2].

Software system

It is important first to consider the time constraints imposed when “real-time” generation of the digital waveforms is attempted in the microcontroller software. The software generation of the waveforms has already been described the maximum frequency of the function generator.

In order to assess the time constraints, on square and quasi-square wave, have one cycle taking minimum time typically is $4\,\mu s$ and maximum value, which is loaded in the timer registers. But for saw-tooth and triangular waveforms, assume that one cycle involves 256 (maximum) increments, 256 (maximum) decrements and 512 comparisons, each update taking typically $3\,\mu s$. The sine wave is defined by 1024 (maximum) sample per cycle, each taking typically $15\,\mu s$ to be accessed from memory and loaded it to the microprocessor, for a Z80 with a 2.5 MHz clock, the maximum frequency from the microprocessor is limited to approximately 12 Hz [3].

To generate the time period in real time it is required that the time counter to be loaded with a value from the CPU corresponding to the required period for square and quasi-square. The rate of decrement of the counter for saw-tooth and triangular waveforms is determined by its clock. When a zero value is reached, the counter output changes and generates an interrupt signal to inform the CPU that the time period has been completed. The CPU responds by loading the next period into the counter and changes the signal from the output port, hence the required waveform is generated.

The software-based system consists of two algorithms, the main program and the interrupt service routine program. The main program initializes mode timers, interrupt registers and other general purpose registers while the interrupt service routine portioned into five subroutines, each one consists of a number of pages, which operates at a certain frequency band generation, that generate its waveform. The software algorithm has the following tasks:

1. Main program, which executes the following, tasks:

Selecting the timer mode of operation.
- Enabling the external interrupt registers.
- Assigning the external interrupt according to the connection of the keypad.
- Performing the do nothing loop statement waiting for pressing any key of the keypad.

2. Interrupt Service Routine program (ISR) which executes the following tasks:

- Scanning the row and column lines of the keypad to determine if a key is pressed.
- Calling a delay function subroutine to allow for a delay time between increment and decrement steps of the keys.
- Selecting the frequency band of generation.
- Displaying the frequency hexadecimal code on the display.
- Running the subroutine of waveform generation.

The software interrupt routine is used for servicing the external interrupt shown in the flow diagram in Figure 2, together with the main program function used to generate the waveforms.

Experimental Results

The experimental results presented in this section were obtained using an experimental microcontroller function generator system. An AT 89C51-version microcontroller was used to provide the waveforms.

The function generator implementation used is based on the microcontroller hardware configuration, which used a crystal oscillator of 12 MHz clock. All the experimental results are delivered from the analogue circuit using the software algorithm described in section 4.

The experimental results have been chosen to illustrate some of the main features of microcontroller which have been discussed previously are shown in Figure 3. The experimental study of these results, which is taken at a wide frequency band, shows the following specifications:

- Quasi-square wave includes:
  - Rise time = 25 nsec.
  - Fall time = 0.1 μs.
  - Overshooting invisible as a single pipe.
  - Ringing invisible.
  - No sag.
  - Different duty cycles.
  - 10 volt peak-to-peak, 0.3 mA output current.
  - Frequency range from less than 1 micro Hz to 250 kHz.

- Square wave includes**:**
  - Rise time = 10 nsec.
  - Fall time = 5 nsec.
  - Neither overshooting nor ringing or sag.
  - 10 volt peak-to-peak, 0.3 mA output current.
  - Frequency range from less than 0.006 Hz to 166.666 kHz, larger that limits can be obtained by using the Quasi-square wave with duty cycle of 0.5.

- Saw-tooth wave includes**:**
  - Frequency range from (1 micro Hz to 150 kHz).
  - 10 & 5 volt peak-to-peak, 0.3 mA output current.

- Triangular wave includes**:**
  - Frequency range from (1 micro Hz to 166 kHz).
  - 10 & 5 volt peak-to-peak, 0.3 mA output current.
  - It can be generated from the saw-tooth algorithm with certain STEP such as 128.

- Sine wave includes**:**
  - Frequency range from (1 micro Hz to 100 kHz).
  - 10 volt peak-to-peak, 0.3 mA output current.

From the experimental waveforms shown in Figure 3, a quasi-square wave generated at 250 kHz, square wave at 160 kHz, saw-tooth at two frequencies; upper frequency at 125 kHz and lower frequency at 50 kHz, triangular wave at 150 kHz and sine wave of upper frequency at 20 kHz and lower frequency at 1200 Hz with 512 lookup table samples.

Therefore, the flexibility of the microcontroller software enables as for generation waveforms continuously at wide frequency band. The correct generation of the frequency wave that corresponds the required signal specifications depends upon the approximate conversion of the decimal time interval to a negative hexadecimal code numbers and the flexibility of choosing the parameters of delay relations when using theoretical calculations.

The frequency fluctuation can be minimized to a very small value by actuate theoretical calculations of the entered hexadecimal code frequency numbers through the keypad.

The linear-mode waveforms suffer from amplitudes decreasing and small distortion when generated above the upper frequency. This is due to least number of STEPs used.

The main limitation of the system is the upper frequency, which is limited by the chip version, the clock frequency of the crystal oscillator, the conversion time of the digital to analogue converter to produce the output signal.

The upper frequency can be increased up to 1.25 MHz for switching-mode waveforms, and up to the range of (250 kHz – 750 kHz) for the linear-mode waveforms, by decreasing the machine cycle time of the 89C51 CPU to less than $1\,\mu s$, which is used. By using a crystal oscillator of 40 MHz and fastest conversion time of the DAC this value can be achieved.

Conclusion

A microcontroller-based function generator is designed and built. The experimental results show that a waveform of acceptable accuracy could be obtained with a high degree of flexibility reduced generator size and a small number of components. The upper frequency range is limited by the switching frequency of the microcontroller.

The fetch and execution time taken by the instruction pointer of the CPU are (1 μs) for each machine cycle, to execute the subprograms. It can be shared by dividing the program into equal fixed-size parts (i.e., each part has the same machine cycle), and to carry out the whole work by performing the tasks of each of its parts by a single microcontroller [7].

By systolic parallel processing technique, the execution time of the subprograms (algorithms) can be shared to be executed by several microcontrollers [7], which operate in parallel, to reduce the busy time of each microcontroller which leads to a shorter time for each function generation and higher frequency output of the system. It also becomes a powerful system of a very wide frequency range and flexibility in choosing the required mathematical functions that are widely used in communications field.

Thus, the future work may include the multiprocessor or multimicrocontroller implementation of the microcontroller-based function generator a step toward a microcontrotroller-based function generator of better specification.

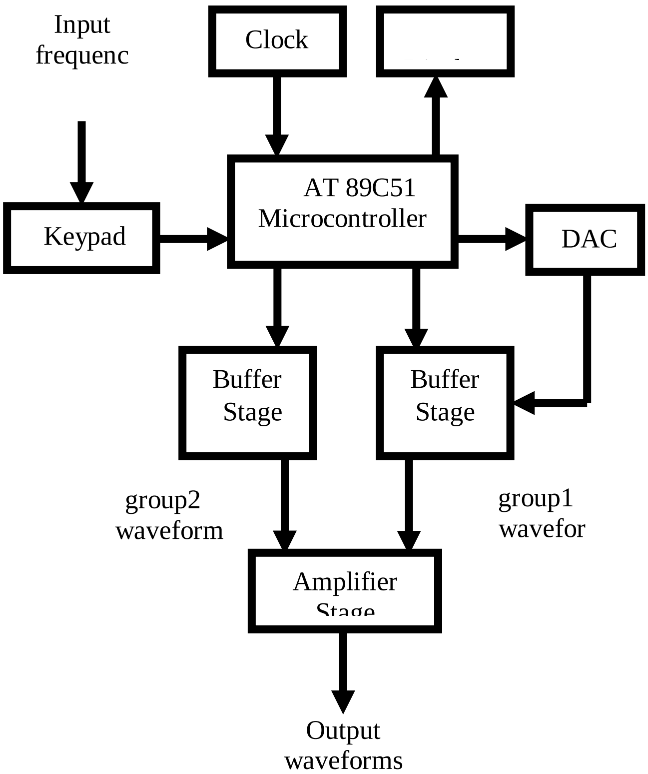
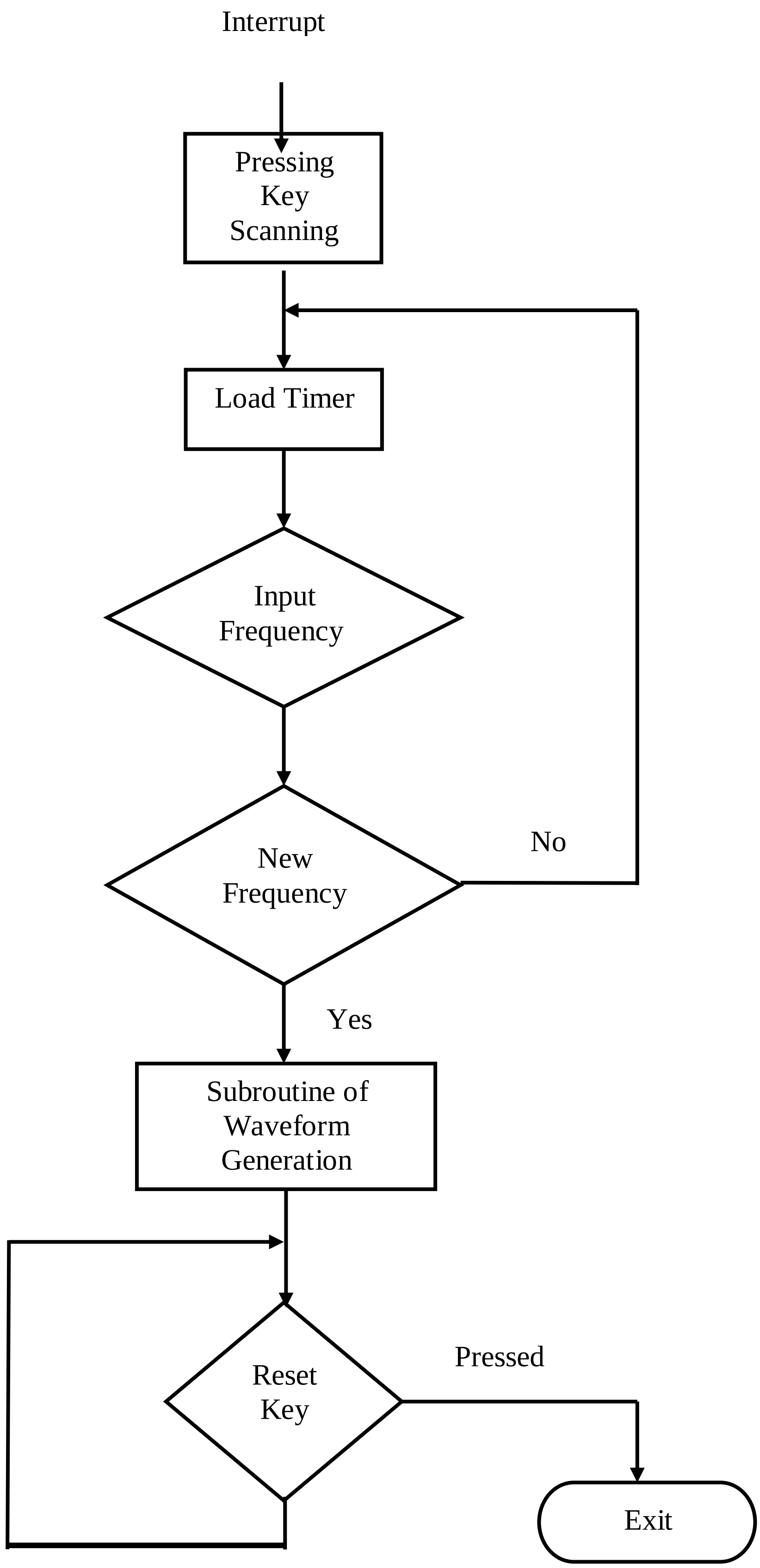
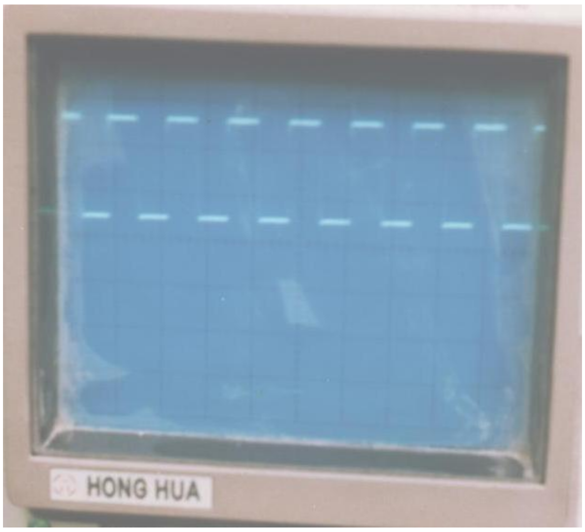


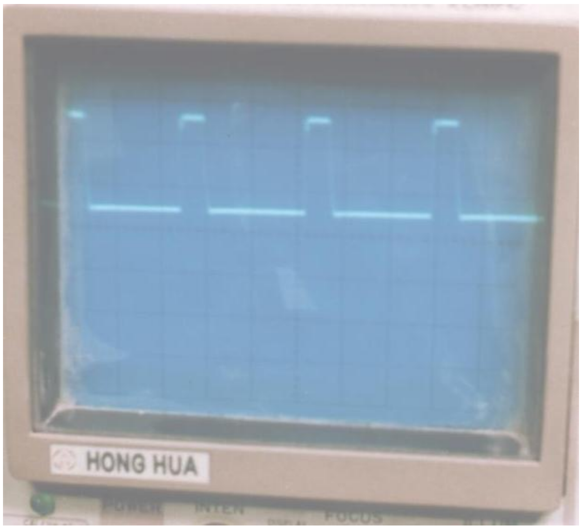
Figure 1: *Hardware-based scheme*



(b): Interrupt Service Routine Program (ISR)

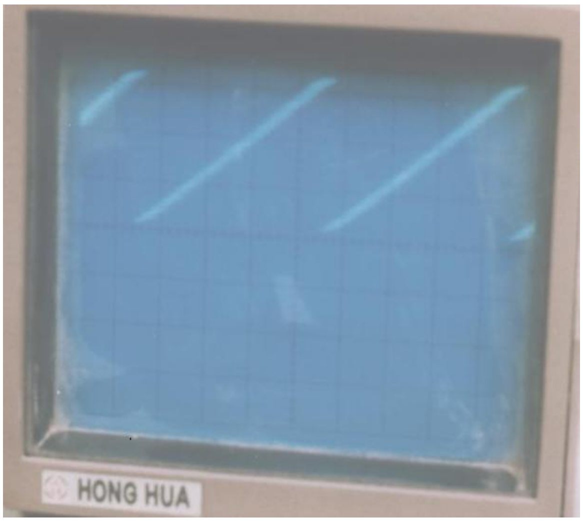
Figure 2: *Continued*

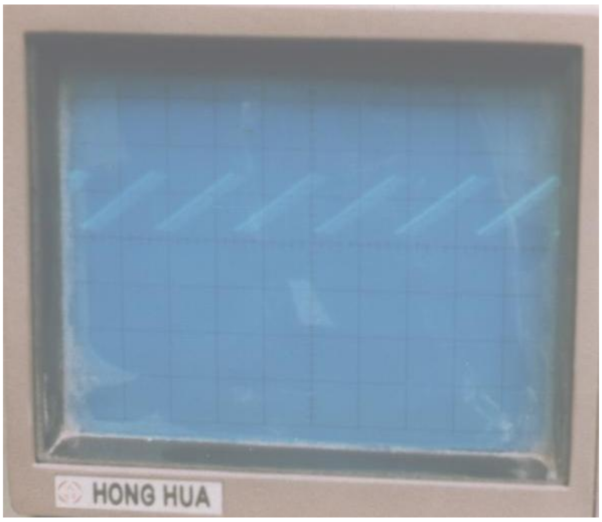




Square Waveform, Frequency =160 kHz

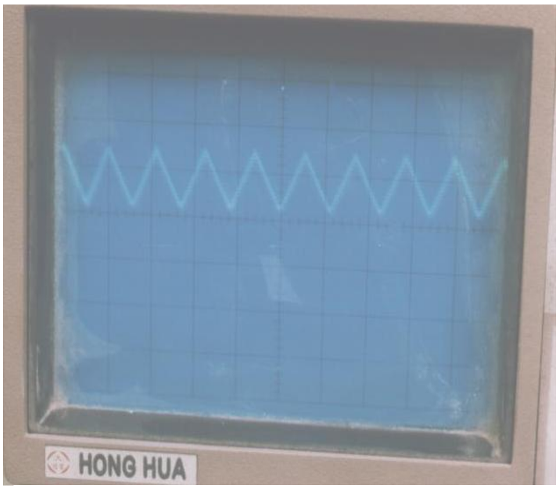
Quasi-Square Waveform, Frequency =250 kHz

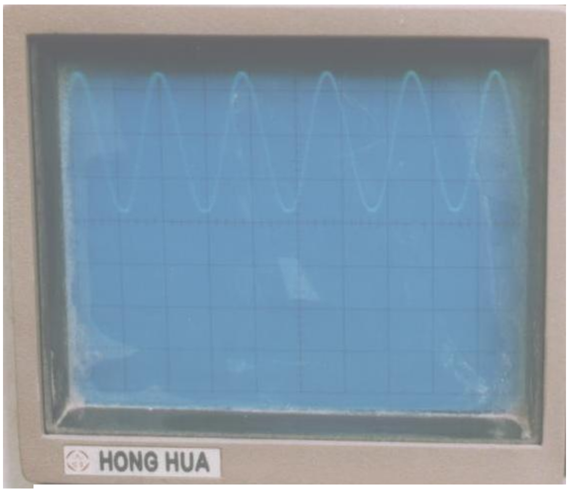




Saw-tooth Waveform, Frequency =50 kHz

Saw-tooth Waveform, Frequency =125 kHz

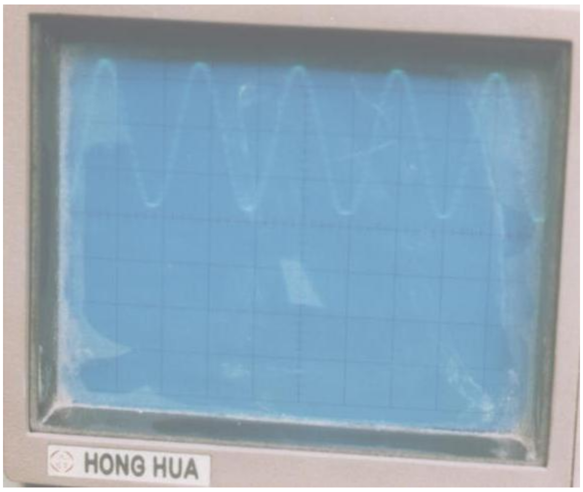




Triangular waveform, Frequency = 150 kHz

Sine wave Frequency = 1200 Hz

Figure 3: *The Experimental results of the microcontroller waveforms*



Sine wave, Frequency = 20 kHz
Figure 3: *Continued*

References

[1] Pei An ،"PC Interfacing using Centronic RS 232 and Game Ports" Newnes, 1998.

[2] S. Yeralanand and A. Ahluwalia, Programming and Interfacing the 8051 Microcontroller, Addison-Wesley, 1995.

[3] S.R. Bowes, C. Eng., M.I. Mech. E., and M.J. Mount, Microprocessor control of PWM inverters." IEE PROC. Vol. 128, pp.293-304, Pt.B, No.6, NOVEMBER 1981 .

[4] S. Mackenzie, The 8051 microcontroller, Prentice-Hall. Inc.,1999.

[5] M. James, Microcontroller Cookbook, Newnes, 1997.

[6] "Intelligent 4 digit dot matrix Alphanumeric LED displays." RS Data Sheet, Po Box 99, Corby, Northants, NN179RS, pp. 232-249, March 1997.

[7] Riyadh A.H. Al-Helali ،"Systolic Algorithms of LMS, RLS Adaptive (FIR) Digital Filters for Adaptive Channel Equalization. " MSc. Thesis, University of Baghdad, Baghdad, Oct. 2005.

مولد ذبذبات يتحكم من متحكم متناهي الدقة (مايكروي)

رياض علي عبد الحسين الهلالي
قسم الهندسة الكهربائية
الجامعة المستنصرية

أ.د. إسماعيل عبد الله محمد
قسم الهندسة الكهربائية
جامعة بغداد

عمار إسماعيل عبد الله
قسم شبكات الحاسبات
وزارة العمل والشؤون الاجتماعية

خلاصة:

تتضمن المقالة وصف لمولد ذبذبات ذو مدى واسع من الترددات تم بناؤه بتحكم من شريحة مسيطر متناهي بالدقة (مسيطر مايكروي) يمكن لمولد الذبذبات توليد موجة جيبيه، ذبذبة مربعة، ذبذبة شبه مربعة، ذبذبة سن المنشار، وذبذبة مثلثية حسب متطلبات مستخدم الجهاز.
بالاستفادة من إمكانيات المعالجة للمسيطر المايكروي تم اختزال كيان المنظومة لدرجة كبيرة.أن الشريحة المنفصلة للمسيطر الدقيق والمستخدمة كأداة لتوليد الأشكال الموجية منحتنا إمكانية التحسين في موثوقية التصنيع، الصيانة وزيادة مرونة السيطرة الرقمية.
لقد تم بناء المنظومة عمليا" وفحصها في عدة مراكز عملية لوزارة الاتصالات وفي قسم الهندسة الكهربائية - جامعة بغداد حيث إن تقييم نتائج الفحص كانت جيدة وأقرت بعض المراكز إمكانية استخدام الجهاز في مختبرات التعليم الأكاديمي والمهني.