



## Birla Institute of Technology & Science, GOA

Submitted By: Group Number 16 2017A7PS0932G ASHWANI SINGH

2017AAPS0348G CHETNA MALHOTRA

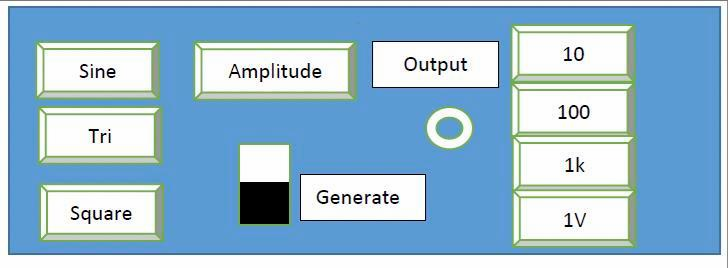
2017A7PS0057G HIMANSHU SHARMA

2017AAPS0248G MUDIT JAIN

2017A8PS0972G SHAGUN TAPADIA

# System to be designed: Frequency Generation

## Description: This system is used to generate a Sine/Triangular/Square waveform of Frequencies ranging from 10 Hz to 99KHz. Voltage is between 0-10V. User Interface:



On system power up the user has to configure the desired type of waveform (square/triangle/square) , frequency and amplitude. To generate a Square Waveform of Frequency 9.35 KHz the user has to press square key, followed by 1K Key- 9 Times, 1K Key – 4 Times, 100 Key –3 Times 10 Key- 5 Times.

To select the Amplitude, the user will have to press Amplitude key and then press the 1V key “n” number of times where “n” is the peak to peak amplitude of the waveform to be generated. (only integer values of output voltages need to be generated) When generate switch should be turned on and then the frequency generation is enabled i.e., the square waveform of that frequency will be generated.

When frequency generation is enabled, if the user wants to change the waveform into another type for e.g. sine he just has to press sine. When a signal of different type/amplitude /frequency has to be generated, the user will have to turn-off the generate switch and then configure the function generator as mentioned above.



* This system is used to generate a Sine/ Triangular/ Square waveform
* The user can select between these three types of waveforms using the keypad.
* The keypad is also used to select the frequency and the amplitude of the waveform generated.
* Frequencies ranging from 10 Hz to 99KHz and Voltage between 0- 10V can be generated by the system.
* The waveform can be changed at a later stage by pressing the button on the keypad.

For example: To generate a Square Waveform of Frequency 9.35 KHz the user has to press square key, followed by 1K Key - 9 Times, 100 Key - 3 Times and 10 Key

* 5 Times.



|  |  |
| --- | --- |
| **COMPONENTS** | **NOS.** |
| 8086 MICRO-PROCESSOR | 1 |
| 8254 Programmable Timer | 1 |
| 74138 3x8 Decoder | 2 |
| 8255 Programmable Peripheral interface | 1 |
| 74LS373 (Octal Latch) | 3 |
| 74LS245(Octal Buffer) | 2 |
| 2732 (ROM – 4k) | 4 |
| 6116 (RAM- 2K) | 2 |
| DAC 0808 | 1 |
| OP.AMP | 1 |
| 3X3 KEYPAD | 1 |
| Digital Oscilloscope | 1 |
| RESISTORS AND CAPACITORS OF REQUIRED VALUE , OR gates , AND gates | |  | | --- | |  | |
| 8259 Programmable Interrupt Controller | |  | | --- | | 1 | |
| 8284 clock generator | |  | | --- | | 1 | |



The following assumptions were made in order to develop the software for the system.

* + At the location FFFF0H, where the instruction pointer points on RESET of microprocessor, there exists a JUMP statement leading to the start of the code.
  + The user can’t press two keys at the same time.
  + The user gives sufficient time between two successive key presses, enough to perform all operations associated with a particular key press. The software however is designed to handle de-bounce.
  + The minimum frequency of signal to be generated is 40Hz and user doesn’t enter anything below this value. This limitation is due to the size of AX register.
  + The maximum frequency of signal to be generated s 99 kHz and user does not enter anything above this value.

### **I/O Map For 8255**

**Base Address:** 00H

### It is I/O mapped I/O System

The addresses of the ports are as follows:

|  |  |
| --- | --- |
| **PORT of 8255 Address** | |
| **PORT A** | 00H |
| **PORT B** | 02H |
| **PORT C** | 04H |
| **Control Register** | 06H |

Data lines: D0-D7 data lines of the microprocessor (as it is connected in even bank) Port Specification:

Group A: Mode 0

Group B: Mode 0

Port A: Output Port B: Not Used

Port C upper: Input Port Port C lower: Output port

Hence, the control word is **10001010b** Which is written to the control register **Address Map**

#### HEX A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **00h** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **02h** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| **04h** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **06h** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

**Address Map for 8254**

Base Address = 08h

Timer Addresses are as follows:

|  |  |
| --- | --- |
| **8254 Timer Address** | |
| **Timer 0** | 08H |
| **Timer 1** | 0AH |
| **Timer 2** | 0CH |
| **Control Register** | 0EH |

Data lines: D0-D7 data lines of the microprocessor (as it is connected in even bank)

Address Map:

#### HEX A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **08h** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **0Ah** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| **0Ch** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| **0Eh** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |

Control Word is **00110100b.**

**Using only counter 0 in mode 2.**

**Address Map for 8259 PIC**

**BASE ADDRESS=** 10H

|  |
| --- |
| **HEX A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **10h** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **12h** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |

**THIS MEANS WE HAVE TO USE A5 , A4 , A3 TO DISTINGUISH BETWEEN THE PERIPHERAL DEVICES.**

# Memory mapping:

## ROM -1 (EVEN) (2732): 00000h-01FFEh

ROM -1 (ODD) (2732): 00001H- 01FFF

RAM-1 (EVEN) (6116): 02000H- 02FFEH

RAM-1 (ODD) (6116): 02001H- 02FFFH

ROM-2 (EVEN) (2732): FE000- FFFFE

ROM -2 (ODD) (2732) : FE001- FFFFF

ACCORDING TO TABLE: A14, A13 used for decoding logic

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A19 | A18 | A17 | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 |
| ROM1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| RAM1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| ROM2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |