Plantilla para Ponencias

Para el XI Encuentro Iberoamericano de Educación

Parra M. Ariel, Batres L. Miguel, Salas P. Diego

4th semester, group A. Center of Basic Sciences, Computer Systems Engineering.

Universidad Autonoma de Aguascalientes

Aguascalientes,Aguascalientes. PC: 20131

{al280862, al350553, al281435}@edu.uaa.mx

**Summary.** This project aims to develop a program in Assembly language for the 8086 processor and DOS 16 bits Operating System, the project aims to generate melodies using the computer's speakers. The program will be using ports 61h to activate and deactivate the speaker and ports 43h and 42h for adjusting the speaker frequency; Preloading four melodies, each at least 30 seconds long, which will be initiated when chosen by the user from a menu; Implementing a piano-like functionality where specific keys trigger corresponding sounds, including flats and sharps. And the program will provide a user-friendly interface for selecting preloaded melodies and enable real-time sound generation akin to playing a piano.

**Key Words:** assembly, intel 8086, DOS.

# Introduction

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

## Comprobar las normas de edición del artículo

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

* Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.
* Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Theoretical Framework

You can communicate with the speaker controller using IN and OUT instructions. The following lists the steps in generating a beep:

* Send the value 182 to port 43h. This sets up the speaker.
* Send the frequency number to port 42h. Since this is an 8-bit port, you must use two OUT instructions to do this. Send the least significant byte first, then the most significant byte.
* To start the beep, bits 1 and 0 of port 61h must be set to 1. Since the other bits of port 61h have other uses, they must not be modified. Therefore, you must use an IN instruction first to get the value from the port, then do an OR to set the two bits, then use an OUT instruction to send the new value to the port.
* Pause (sleep) for the duration of the beep.
* Turn off the beep by resetting bits 1 and 0 of port 61h to 0. Remember that since the other bits of this port must not be modified, you must read the value, set just bits 1 and 0 to 0, then output the new value.

## Encabezados y tablas

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

**Tabla 1.** Tamaño de letra en cada tipo de encabezado. El epígrafe o título de la tabla siempre debe ir encima de ésta, centrado si sólo ocupa una línea y justificado si se extiende a más de una (como en este ejemplo), y debe acabar en punto.

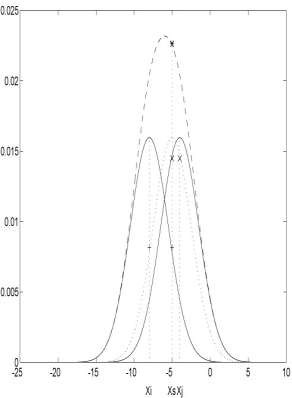
Encabezado Ejemplo Tamaño y Estilo Título (centrado) **Título del Trabajo** 14 puntos, negrita

|  |  |  |
| --- | --- | --- |
| 1er nivel | **1 Introducción** | 12 puntos, negrita |
| 2o nivel | **2.1 Encabezados** | 10 puntos, negrita |
| 3er nivel | **Lema 3.** Sigue Texto … | 10 puntos, negrita |
| 4o nivel | *Observación.* Sigue Texto … | 10 puntos, cursiva |

**Lemas, Enunciados y Teoremas.** Deben estar numerados consecutivamente por orden de aparición en el texto, por ejemplo Lema 11.

## Figuras

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



**Fig. 1.** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

## Frecuency Number used in beep macro

The frequency number is a word value, so it can take values between 0 and 65,535 inclusive. This means you can generate any frequency between 18.21 Hz (frequency number = 65,535) and 1,193,180 Hz (frequency number = 1).

Knowing this bounds and using the known frequencies in Hertz of musical notes,we created formulas to convert these frequencies into the corresponding frequency numbers. These frequency numbers can then be used as inputs to generate the desired musical tones using the beep macro found in mac.inc

When you divide the given frequency by the maximum frequency, you get a ratio that indicates how many times the given frequency fits into the maximum frequency value and since frequency numbers start from 0, we subtract 1 to ensure that the highest frequency (1,193,180 Hz) maps to the highest frequency number (65,535), rather than 65,536.

The difference represents how far the frequency number is from the highest possible frequency. Then we divide the difference by the frecuency number itself because as the frequency number decreases, the ratio increases, resulting in a higher frequency. Finally by multiplying the ratio by smallest frequency value (18.21 Hz), we are essentially scaling it to fit within this range.

## Fragmentos de código

Los fragmentos de código tendrán como fuente Courier de tamaño 9 puntos. Irán precedidos de una explicación de tamaño 10 puntos. Deben comenzar por Algoritmo, y el número correspondiente de manera secuencial (encabezado de nivel 3) además del título. Ejemplo:

**Algoritmo 1.** Ejemplo de un programa de ordenador, sacado de Jensen K., Wirth N. (1991) *Pascal user manual and report. Springer*, New York.

program Inflation (Output)

{Assuming annual inflation rates of 7%, 8%, and 10%,... years};

const MaxYears = 10; var Year: 0..MaxYears;

Factor1, Factor2, Factor3: Real;

begin

Year := 0;

Factor1 := 1.0; Factor2 := 1.0; Factor3 := 1.0;

WriteLn('Year 7% 8% 10%'); WriteLn; repeat

Year := Year + 1;

Factor1 := Factor1 \* 1.07; Factor2 := Factor2 \* 1.08; Factor3 := Factor3 \* 1.10;

WriteLn(Year:5,Factor1:7:3,Factor2:7:3,

Factor3:7:3) until Year = MaxYears

end.

## Citas y referencias

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Development

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Results

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Conclusions

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Bibliography

1. Richard R, Eckert. (2007). *The Intel 8253/8254 Programmable Interval Timer and Sound on a PC*. Binghamton University. Retrieved from <https://www.cs.binghamton.edu/~reckert/220/8254_timer.html>
2. Andezuthu D, Murugan. (2009). *8086 Assembly Language Program to Play Sound Using PC Speaker.* Retrived from <http://muruganad.com/8086/8086-assembly-language-program-to-play-sound-using-pc-speaker.html>
3. Evans, David. (2022). *x86 Assembly Guide.* University of Virginia. Retrived from <https://www.cs.virginia.edu/~evans/cs216/guides/x86.html>
4. Cardoso, G. Célio. (2015). *Internal Speaker*. Universidade Estadual de Campinas. Retrived from <https://www.ic.unicamp.br/~celio/mc404s102/pcspeaker/InternalSpeaker.htm>
5. Ruth, Anderson. (2017). *Assembly Programming IV*. University of Washington. Retrived from <https://courses.cs.washington.edu/courses/cse351/17sp/lectures/CSE351-L10-asm-IV_17sp-ink.pdf>
6. William R, Mark. (n.d)*. x86 Control Flow*. University of Southern California. Retrived from [https://ee.usc.edu/~redekopp/cs356/slides/CS356Unit5\_x86\_Control](https://ee.usc.edu/~redekopp/cs356/slides/CS356Unit5_x86_Control#page138)
7. Hyde, Randall. (1996). *The Art of Assembly Language*. Retrived from <https://www.randallhyde.com/AssemblyLanguage/www.artofasm.com/DOS/pdf/0_AoAPDF.html>
8. Caldwell, Urtis. (2004). *Intel 8086 Instruction Timing*. Methodist College. Retrived from <https://www.oocities.org/mc_introtocomputers/Instruction_Timing.PDF>
9. Craig Stinson, Brad Kingsbury, et al. (2001). *The Assembly Language: database INT 15h, 86h (134) Wait XT-286, AT*. Retrived from <http://vitaly_filatov.tripod.com/ng/asm/asm_026.13.html>
10. Suits H, Bryan. (2022). *Physics of Music - Notes*. Michigan Technological University. Retrived from <https://web.archive.org/web/20220124125158/https://pages.mtu.edu/~suits/notefreqs.html>