# TRIBHUVAN UNIVERSITY

Institute of Engineering
Pulchowk Campus
Department of Electronics and Computer Engineering

A Graphics Project Report
On
# "Bloxorz"

**Submitted By:**
Amrita Poddar (063BCT504)
Kapila Rajbhandari(063BCT513)
Ritu Kandoi (063BCT528)

**Submitted to:**
Department of Electronics and Computer Engineering
Pulchowk Campus

23/01/2011

# ABSTRACT

Computer Graphics is one of the powerful tools for the rapid and economical production of pictures. It remains one of the most exciting and rapidly growing computer fields. Actually it is the field related to generation of graphics using computers. It includes the creation, storage and manipulation of images of object.

The objects that are to be manipulated by Computer Graphics come from diverse fields viz physical, mathematical, engineering, architectural, abstract structures and natural phenomenon.

There is the vast field related to Computer Graphics. Some of the applications of Computer Graphics are as follows:-

**Design of engineering and Architectural System**; **Presentation Graphics**, to produce illustrations for reports or to generate slides or transparencies for use with projectors; **Computer Art**, artists use computer graphics such as paintbrush; **Entertainment ,** used in making motion pictures, music videos and other shows ; **Education and Training**, computer-generated models of physical, financial, and economics system are often used as educational aids ; **Visualization.**

Our project is also related to the applications of Computer Graphics. We have made an attempt to represent the basic interface of a game named BLOXORZ using Computer Graphics. The basic concept of the project is to work on hidden surfaces, data entries, and rotation of the object.

# ACKNOWLEDGEMENT

We express our gratitude to the Department of Electronics and Computer Engineering, IOE, Pulchowk Campus for providing us with the opportunity to work on a Graphics Project as a part of our syllabus. We became more familiar with the computer graphics while working on this project. We are very grateful to **Mr. Anil Verma**, a faculty member of our department and our project supervisor, for giving us guidance and his valuable suggestions regarding this project.

Also we extend our gratefulness to our colleagues and friends who are directly or indirectly linked with our project and for their help in every step when there arisen any kinds of problems by giving different ideas, concepts on the related topics. There are still other people who have helped us directly or indirectly, we are to blame if we forgot to mention your name in this acknowledgement.

We would highly appreciate any constructive comments, suggestions and criticism from colleagues and the teachers.

# **PREFACE**

Modern world is crawling towards the advanced technology. No doubt, it is the extensive use of computer which brings the civilization towards the modernism. Within the different facets of computer, Graphics is one of the vital fields of computer applications and it makes possible for amazing pictorial realization of physical   and abstract things.

Since we are not the professional people now, we concentrate mainly to implement more graphical concepts we have studied during the semester course (Lecture classes and practical labs). So, we come with the output of 3D object with different attributes and geometric features. Seemingly, it is a small work but will be the gateway to cope with advanced graphics technology.

# TABLE OF CONTENTS

# **INTRODUCTION**

Bloxorz is a simple idea for a puzzle and skill game that is beautifully executed. The objective is to tumble a rectangular block through each stage and deposit it into the square hole at the end of each stage.

To move the block around the world, the Left, Right, Up, and Down arrow keys are used. One should be careful not to fall off the edges; the game level will be ended if this happens.

All the faces of the block and the platform are drawn using simple drawing algorithms. We use Back Face Detection to view the hidden surfaces while performing various transformations.

The main objective of our project is to simulate a game named BLOXORZ in 3D environment using computer graphics techniques learnt in our course. We showed various transformations, detected the visible surfaces, and also found out the intensity of each surfaces with respect to the viewing plane.

# Application Development Environment

We used Java 1.6 (jdk1.6.0_02) programming language to create the rectangular blockand the base platform. Swing (which comes bundled with JDK) was used for User Interface (UI) portion of the project. We used Net beans 5.5.1 IDE for the project development. Code collaboration between the two developers was done using the subversion (svn) repository provided by google-code (http://code.google.com).

On the initial phase of our project we studied various object modeling techniques. Then we designed the wireframe structure of the object that we model. Then using the java AWT (Abstract Window Toolkit) we drew the objects that we modeled. Then we implemented the basic transformation for the object. Beside that the GUI for Input and Output is based on java Swing API.

NetBeans refers to both a platform framework for Java desktop applications, and an integrated development environment (IDE) for developing with Java, JavaScript, PHP, Python, Ruby, Groovy, C, and C++. The NetBeans IDE is written in Java and runs everywhere where a JVM is installed, including Windows, Mac OS, Linux, and Solaris. A JDK is required for Java development functionality, the PHP bundle can be used with just the JVM.

The NetBeans Platform allows applications to be developed from a set of modular software components called modules.

# LITERATURE REVIEW

In our project we applied the theoretical concept of 3D-geometric transformations, and different illumination models and perspective projection technique. In three dimensional geometric transformations, we used:

- Translation

- Rotation

## Translation:-

A translation is applied to an object by repositioning it along a straight path from one coordinate location to another. This is a rigid body transformation. We translate a three dimensional point by adding translation vector. In the three dimensional homogeneous coordinate representation, a point is translated from position p(x,y,z) to position p'($x_1$,$y_1$,$z_1$) with the matrix operation:

$$\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

## Rotation:-

It is the transformation that repositions an object along the circular path through the given angle and point of rotation (pivot point).

We used the transformation about x axis, y axis and z axis using the respective matrix operation. The matrix along with the translational matrix shift the reference point of the object to the origin is used and after rotation again the object is translated to the original position by using another translational matrix.

For rotation about z-axis:

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

For rotation about x-axis:

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

For rotation about y-axis:

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

## VISIBLE SURFACE DETECTION

Since we have used the polygons and thus it will be easier to calculate the plane equation of the polygons and the normal vector can be directly calculated from the plane equation. Therefore we have chosen the Back-Face Detection algorithms for visible surface detection or removal of hidden surface.

A simple object-space method of for identifying the back face of a polyhedron is based on the back face detection method. It Compares the dot product of the Normal Vector of the Surface(**N**) with the Normal Viewing Vector(**V**).

If **N.V**>0 then the surface is visible else the surface is not visible.

The equation for plane surface can be expressed in the form of

$$Ax + By + Cz + D = 0$$

The calculation of the plane coefficients can be done from the following equations if $(x_1,y_1,z_1)$, $(x_2,y_2,z_2)$ and $(x_3,y_3,z_3)$ are vertices of the plane

$$A = y_1(z_2 - z_3) + y_2(z_3 - z_1) + y_3(z_1 - z_2)$$

$$B = z_1(x_2 - x_3) + z_2(x_3 - x_1) + z_3(x_1 - x_2)$$

$$C = x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)$$

$$D = -x_1(y_2z_3 - y_3z_2) - x_2(y_3z_1 - y_1z_3) - x_3(y_1z_2 - y_2z_1)$$

# Projection:

Projection is the important part of any three dimensional geometry. In our project, we are applying the perspective projection in which we transformed the points along projection lines that meet at the projection reference point. We have set the projection point at position zprp along the zv axis and we place the view plane at zvp.

Thus, we calculated the projected points as follows:

xp=x(zprp-zvp/zprp-z)=x(dp/zprp-z)

yp=y(zprp-zvp/zprp-z)=y(dp/zprp-z)

## Illumination Model:

The next features we added to our graphical object are illumination modeling and surface rendering. For the illumination model, we combined the effect of ambient light, diffuse reflection and specular reflection. The expression for the combined effect of diffuse and specular reflection in the illumination model is:

$$I = K_a I_a + K_d I_l (N.L) + K_s I_l (N.H)^{ns}$$

Where

$K_a$: ambient reflection coefficient

$I_a$: the incident ambient intensity

$I_l$: the intensity of the light source

$K_d$: the diffuse reflection coefficient

$K_s$: the colour-independent specular coefficient

N: the surface normal (unit vector)

L: the direction of light source (unit vector)

V: unit vector towards viewer

$n_s$: the specular reflection parameter

H: the unit bisector of angle between L and V.

$$H = L+V /|L+V|$$

# LIMITATIONS AND FURTHER ENHANCEMENTS

The program seems to be well complete but there are many things that need to be improved.

In the visible surface detection, z-coordinates are determined, sorted before drawing plane surfaces of the block. The program does not actually use any rendering models. The program can be made to use rendering models to produce more realistic scene. Also the block lacks shading so that any slanting surface is having constant light. The block and platform can be shaded such that any slanting surface is well shaded. The game could be enhanced by creating a boundary around the platform and adding more levels to make it interesting.

Though the program uses 3D calculation, it could be improved to effectively calculate using matrix calculation. Use of matrix would simplify many calculations. It also does not use any vector calculation; though vector calculation is complex, it will make the program more robust; the lighting, shading effects can easily be applied using vector calculations.

The program could have used effective lighting models that can be easily moved around the client screen so that the game's interface is more effectively presented.

# **SAMPLE OUTPUT**

# **CONCLUSION**

With the completion of this project we learnt the project management techniques. This project though does not perform any advanced task but proved an important step for learning the basic programming techniques used in computer graphics.

We have finished this project successfully although there were some problems retrieved during the completion of the project. At initial stage, we were much unfamiliar with computer graphics and its wide application, but after the completion of this project, we became very much familiar and gained a lot of knowledge and several techniques of Computer Graphics of 2D and 3D objects. The concept of three dimensional transformations became very much clear and we were able to translate and rotate the block easily in the required directions. The Back Face Detection algorithm is used to find the visible surfaces.

# <u>REFERENCES</u>

1. Computer Graphics- C Version

    Donald Hearn and M.Pauline Barker

2. Computer Graphics-principles and practice

    Foley, vanDam, Feiner, Hughes

3. How To Program In Java

    Deitel and Deitel