Simulating Dendrimer Growth and Light Harvesting

By

Angel Vallejo
Department of Chemical Engineering
University of Illinois at Chicago
810 S. Clinton Street, Chicago, IL 60607-7000 USA
avalle3@uic.edu

Abstract

In this project I started studying about dendrimers, their chemistry and their physics. After studying the dendrimer and its mechanism in light harvesting, the focus turned into simulating it. Using Visual Basic's graphical capabilities, I have designed some programs to simulate the growth of the dendrimers. This will allow the user to easily understand and visualize the characteristics that make the dendrimer unique.

Introduction

Dendrimer chemistry was first introduced in 1978 by Fritz Vögtle and coworkers [1]. He synthesized the first "cascade molecules", today known as dendritic molecules. In 1985, Donald A. Tomalia, working in the field of polymer chemistry, synthesized the first family of dendrimers [13]. These contributions to the field have paved the way for continuing research in this promising area.

During the past couple of years, there has been an intensifying interest in dendrimer chemistry. The word "dendrimer" originated from two words, the Greek word dendron, meaning tree, and meros, meaning part. A dendrimer (Figure 1) is a synthetic, man-made, 3-dimensional macromolecule, which is prepared in a series of repetitive reactions from simple branched monomer units [2]. The synthesis used for dendrimer preparation permit almost entire control over the critical molecular design parameters such as size, shape, surface/interior chemistry, flexibility, and topology [2].

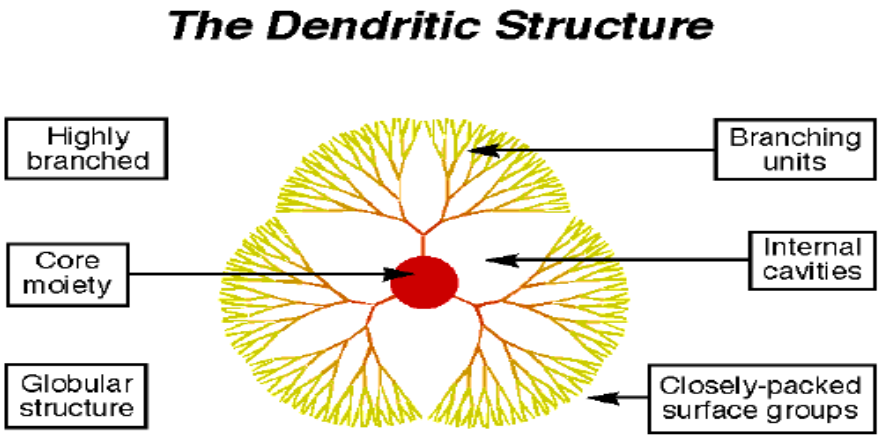


Figure 1: A diagram of a dendrimer showing its unique structural characteristics. This figure was copied from [15].

This leads to molecules that have the exact required structure, composition, and molecular weight. The dendrimer's unique chemical structure, the way all chemical

bonds can be accurately described, and its high degree of symmetry is what separates it from other hyper-branched polymers. This is also what makes it the ideal building block for the next industrial revolution, which is nanotechnology [3].

Synthesis:

There are two major synthetic procedures in which dendrimers can be prepared. They are divergent and convergent synthesis (Figure 2). In divergent synthesis, the synthesis starts at the core and works its way out to the periphery. The core consists of multiple reaction sites. This core is treated with an excess of the first monomer reacting with all the cores reaction sites. This monomer also has reactive groups that are ready to react. An excess of a second monomer is reacted with the half generation (core and monomer), giving rise to the first generation. A continuation of this iterative reaction procedure leads to the second and third generations where none of the unique characteristics are yet noticeable.

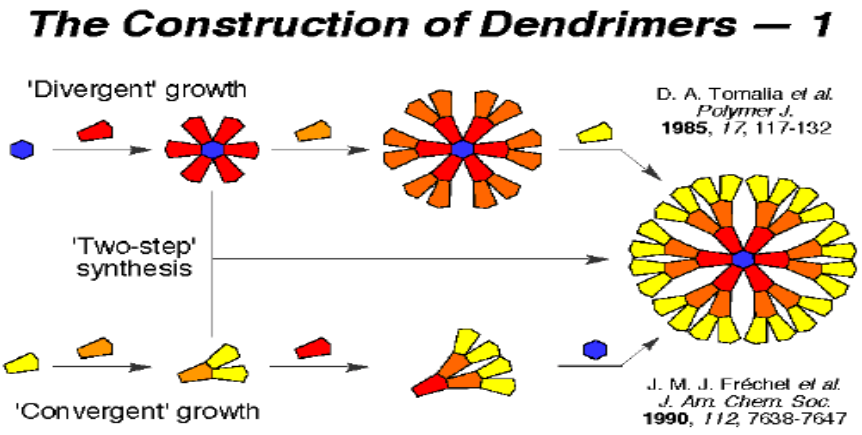


Figure 2: Schematic representation of Divergent and Convergent growth. This figure was copied from [15].

It is not till the fourth generation where the dendrimer becomes a highly structured sphere. Beyond the fifth generation steric overcrowding could occur, and prevent complete reaction of the molecule. This also damages the dendrimers uniform structure and leads to defects inside the molecule. In convergent synthesis these problems are avoided by starting synthesis at the peripheral and concluding at the core. Here dendron segments are constructed and then assembled around the core. Only two reactions are required at each step of the growth process, and not simultaneous reactions creating a uniform final product. Convergent synthesis makes it easier to yield the desired dendrimer and have complete control over all molecular design parameters [12].

Light Harvesting:

The dendrimer uniqueness makes it the building block for many potential applications. Such applications include light harvesting, drug delivery, catalysis, data storage, chemical sensors, and more. Light harvesting is the trapping of energy via peripheral chromophores and funneling to a central point where it is converted back into visible light. The dendrimer possesses the properties that facilitate such a conversion. These properties include its tree-like structure that acts as an energy gradient for the funneling of energy. The large amount of absorbing units on the periphery, gives the high probability of capture of light. The relatively short distance from the periphery to the core allows for high efficiency energy transfer [4]. This high efficiency has been shown to be as high as 98% in a phenylacetylene dendrimer as done by Jeffrey S. Moore and coworkers [5,8].

The main goal in researching light harvesting in dendrimers was to understand the mechanism involved. The mechanism begins with the periphery chromophore molecules capturing the energy of photons from light. These photons excite the electrons in the molecules and raise them from their ground state to their excited state. Interchromophore energy transfer then occurs in one of two ways, Dexter excitation transfer, or Forster excitation transfer. In Dexter excitation transfer the energy is transferred through-bond electron exchange. This electron exchange requires a strong donor to acceptor orbital overlap and is therefore a short-range interaction (<10 Å) according to [6]. In Forster excitation transfer the energy is transferred through-space dipole-dipole interaction. In this case, the donor to acceptor orbital overlap is not necessary, allowing the chromophores to be separated by larger distances (10-100 Å) [6]. Depending on the monomers used to synthesize the dendrimer that will affect the energy transfer mechanism utilized. Using any of the above energy transfer mechanisms, the energy is channeled to the core where it is converted into visible light. Refer to figure 3 for flow chart representation of this mechanism.

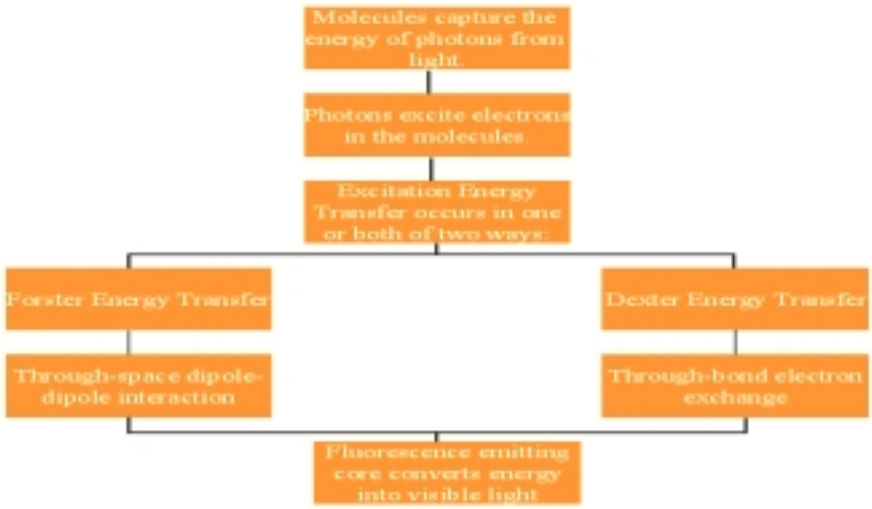


Figure 3: Flow chart representation of the light harvesting mechanism.

Procedure and Modeling

In order to better represent the growth in dendrimers, I have developed a couple of computer programs. All these programs were done in the computer programming language of Visual Basic. The computer programs capture the complexity of dendrimer growth. The programs developed during this project are included in Appendix A of this report.

The first program, *Dendrimer_Growth1*, the user has the ability to input the generation desired for viewing. A 2-dimensional representation of the dendrimer is displayed. As the user enters a low generation number (0-3), the dendrimers structural characteristics will not be yet seen. As a higher generation number is entered (4-7), the structural characteristics of a dendrimer begin to take shape. The program will only show up to generation 7 because this is where steric crowding occurs. Any request for a higher generation number than the 7^{th}, will default to the 7^{th} generation.

The second program, *Dendrimer_Growth2*, displays the convergent growth of the dendrimer. Starting at the core, branches grow exponentially until they become so densely packed that a globular conformation is formed. It is here where no further growth is allowed and the program automatically restarts itself back at the core. This program will continue to run until it is terminated.

The third program, *Dedrimer_Growth3*, allows the user to select a specific core of a dendrimer from a drop down menu. The selection will then execute the divergent growth of the selected core. Once steric overcrowding occurs, the program will stop to show that stage. In addition, the user has the flexibility to reselect any other available cores in the drop down menu. The available cores listed in the drop down menu are NH_3,

N_2H_4, $C_4N_2H_{12}$, and $C_2N_2H_8$. Each core was reacted in a divergent approach to produce the simulated growth. NH_3, N_2H_4, and $C_2N_2H_8$ are all reacted with acrylic acid followed by the amidation of ethylenediamine [1]. The core molecule, $C_4N_2H_{12}$, is reacted with acrylonitrile followed by a reduction to yield the initial functional groups [1]. It should be noted that nowhere in the program are reactions shown, only the stages that follow such reactions. Figures 4-6 show the results of running programs.

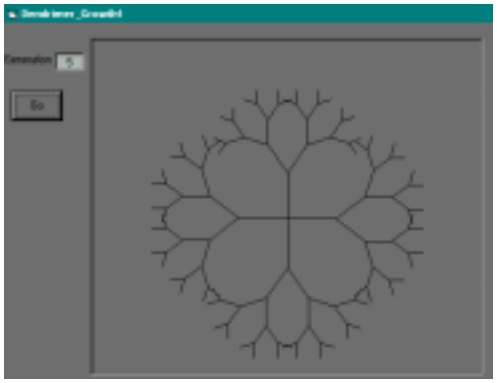


Figure 4: Shows the result of an input of 5 by the user in the program *Dendrimer_Growth1*.

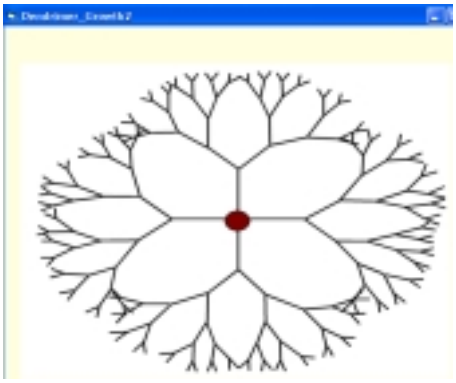


Figure 5:A stage of the simulated growth of the dendrimer as shown in the program *Dendrimer_Growth2*.

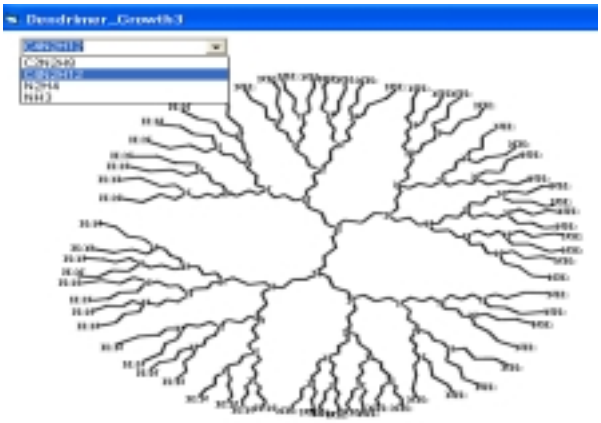


Figure 6: The last stage of the simulated growth of a polypropylene imine dendrimer as shown in the program *Dendrimer_Growth3.*

Conclusion and Recommendations

Throughout this research project the aim has been to gain understanding of the dendrimer and how it grows. The intent of the work presented here is to allow inexperienced readers to fully grasp the complexities of the dendrimer. The programs were created to facilitate the understanding of dendrimer growth via visual simulation. These programs will make the simulation of the dendrimer more palpable.

This research presents avenues for further research. The program *Dendrimer_Growth3* could be further developed to include cores other than the four mentioned. A program could be designed to predict light harvesting capabilities of dendrimers. Prediction of the efficiency of energy transfer form the peripheral to the core in the dendrimer could be the subject of another program.

Acknowledgements

The author would like to acknowledge the NSF-REU site for support in this research and also for providing research experience. He would also like to thank G.A. Mansoori for his guidance throughout this research.

References

1.E. Buhleier, W. Wehner, F. Vögtle, *"Cascade and Nonskid-chain-like Synthesis of Molecular Cavity Topologies"*, Synthesis, (2), 155-158, 1978, see also *"Dendrimer"*, F. Vögtle (Editor), Springer-Verlag, Berlin, 1998

2.T. Cagin, G. Wang, R. Martin, and W.A. Goddard, *"Molecular Modeling of Dendrimers for Nanoscale Applications"*, Nanotechnology, 11(2), 77-84, 2000

3.G.A. Mansoori, *"Advances in Atomic and Molecular Nanotechnology"*, Nanotechnology: The Emerging Cutting-Edge Technology, UN-APCTT Tech Monitor, Sep-Oct 2002 Special Issue

4.A. Bar-Haim, and J. Klafter, *"Dendrimers as Light Harvesting Antannae"*, Journal of Luminescence, 76 & 77, 197-200, 1998

5.J.S. Moore, P. Bharathi, and C. Devadoss, *"Energy Transfer in Dendritic Macromolecules: Molecular Size Effects and the Role of an Energy Gradient"*, Journal of American Chemical Society, 118(40), 9635-9644, 1996

6.A. Adronov, and J.M.J. Frechet, *"Light Harvesting in Dendrimers"*, Chemical Communications, 1701-1710, 2000

7.J.M.J. Frechet, and D.A. Tomalia, *"Dendrimers and Other Dendritic Polymers"*, John Wiley & Sons, Inc., New York, NY, 2001

8.S.F. Swallen, Z. Shi, W. Tan, Z. Xu, J.S. Moore, and R. Kopelman, *"Exciton Localization Hierarchy and Directed Energy Transfer in Conjugated Linear Aromatic Chains and Dendrimeric Supermolecules"*, Journal of Lumimescence, 76 & 77, 193-196, 1998

9.R. Stephens, *"Visual Basic Graphics Programming"*, John Wiley & Sons, Inc., New York, NY, 2000

10.W.S. Freeze, *"The Visual Basic 5 Programmer's Reference"*, Ventana Communications Group, Research Triangle Park, NC, 1997

11.B. Overland, *"Visual Basic 6"*, IDG Books Worldwide, Foster City, CA, 1999

12.S.M Grayson, and J.M.J. Frechet, *"Convergent Dendrons and Dendrimers: From Synthesis to Applications"*, American Chemical Society, 101(12), 3819-3868, 2001

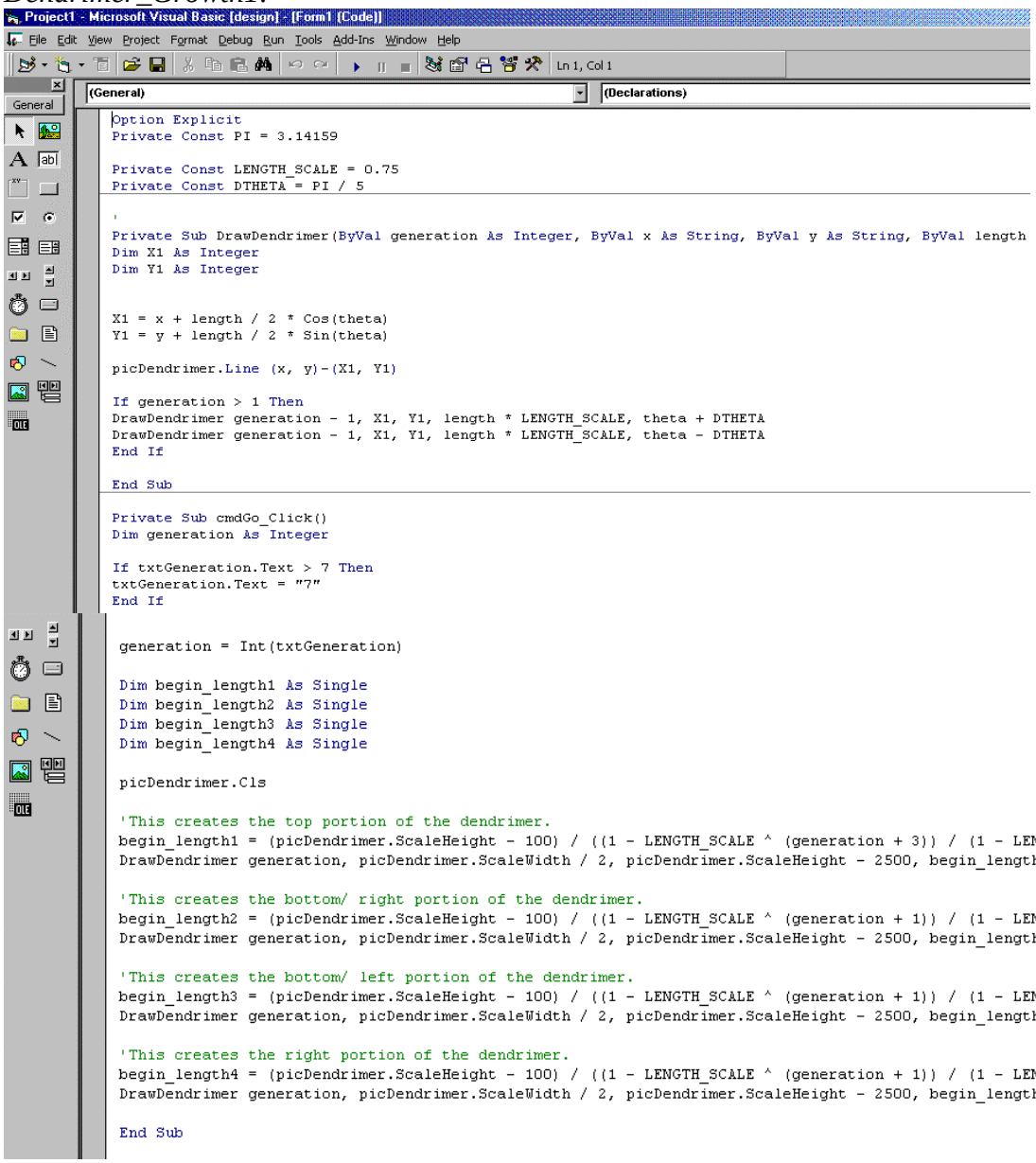
13.D.A. Tomalia, H. Baker, J. Dewald, M. Hall, G. Kallos, S. Martin, J. Roeck, J. Ryder, P. Smith, *"A New Class of Polymers: Starburst-Dendritic Macromolecules"*, Polym. J., 17(1), 117-132, 1985

14.V.D. Kleiman, J.S. Melinger, D. McMorrow, *"Ultrafast Dynamics of Electronic Excitations in a Light-Harvesting Phenylacetylene Dendrimer"*, J. Phys. Chem. B, 105(24), 5595-5598, 2001

15.A.N. Shipway, URL: http://www.ninger.com/dendrimer/, 1997

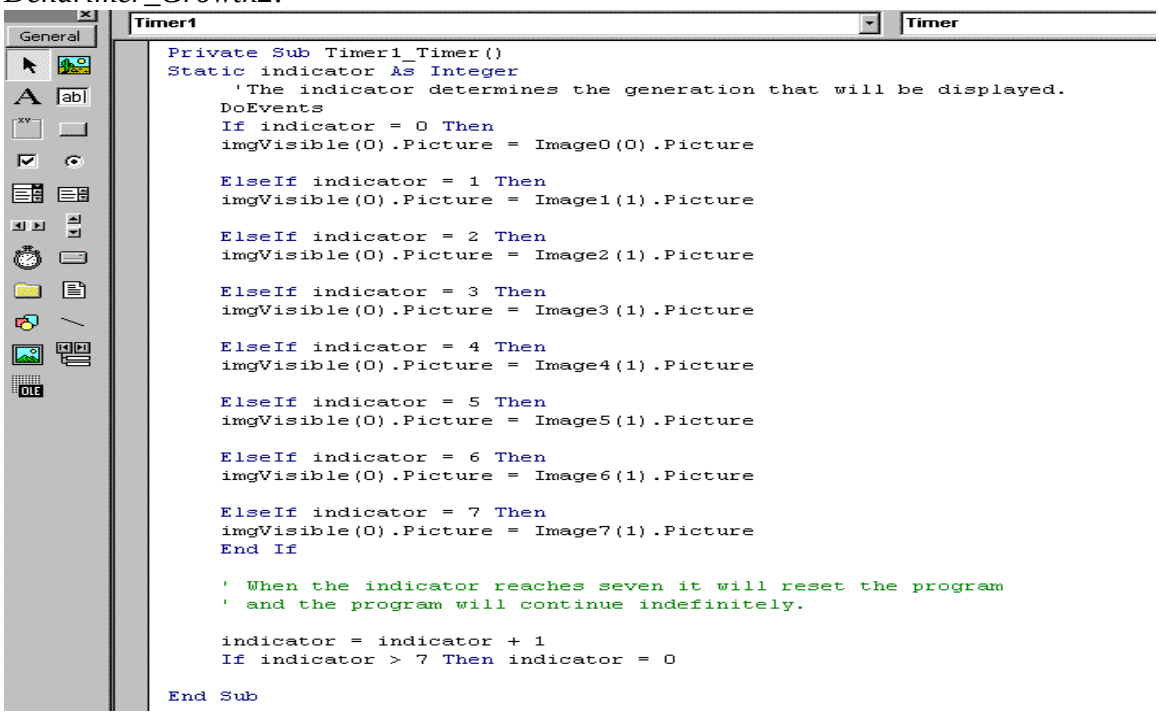
Appendix A

Dendrimer_Growth1:



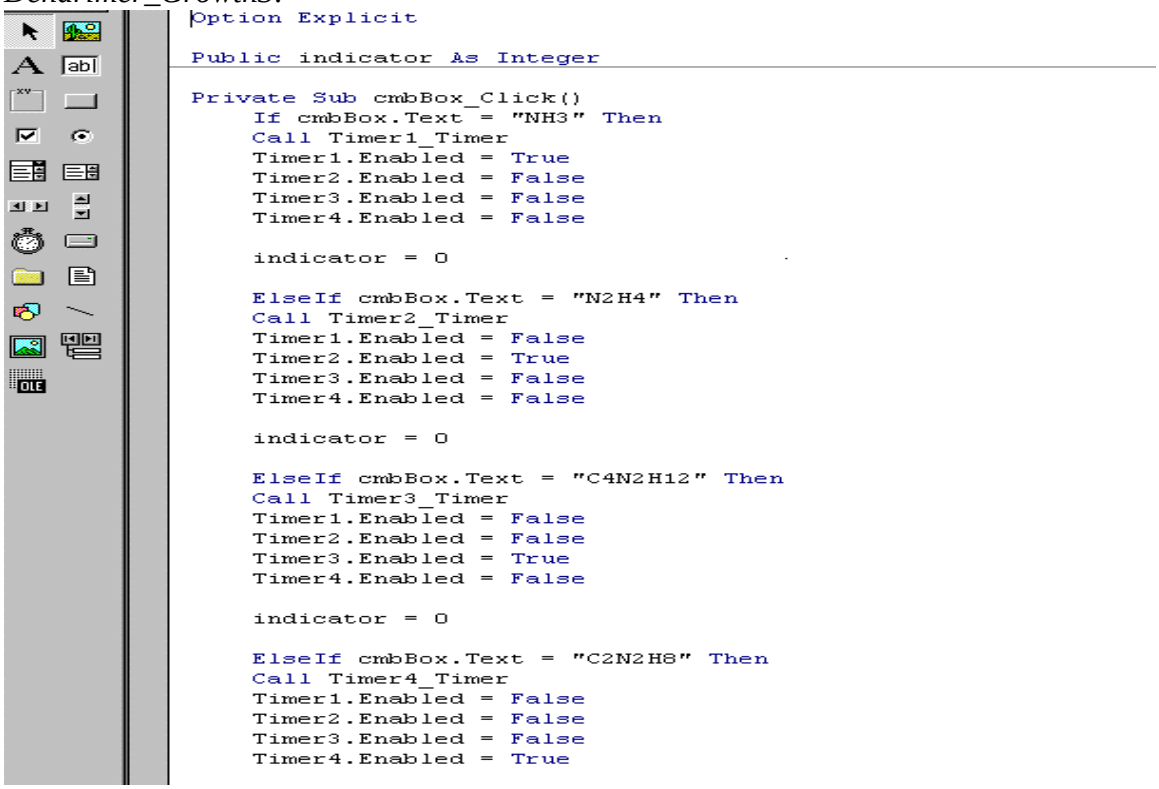

```
Option Explicit
Private Const PI = 3.14159

Private Const LENGTH_SCALE = 0.75
Private Const DTHETA = PI / 5

'
Private Sub DrawDendrimer(ByVal generation As Integer, ByVal x As String, ByVal y As String, ByVal length
Dim X1 As Integer
Dim Y1 As Integer


X1 = x + length / 2 * Cos(theta)
Y1 = y + length / 2 * Sin(theta)

picDendrimer.Line (x, y)-(X1, Y1)

If generation > 1 Then
DrawDendrimer generation - 1, X1, Y1, length * LENGTH_SCALE, theta + DTHETA
DrawDendrimer generation - 1, X1, Y1, length * LENGTH_SCALE, theta - DTHETA
End If

End Sub

Private Sub cmdGo_Click()
Dim generation As Integer

If txtGeneration.Text > 7 Then
txtGeneration.Text = "7"
End If


generation = Int(txtGeneration)

Dim begin_length1 As Single
Dim begin_length2 As Single
Dim begin_length3 As Single
Dim begin_length4 As Single

picDendrimer.Cls

'This creates the top portion of the dendrimer.
begin_length1 = (picDendrimer.ScaleHeight - 100) / ((1 - LENGTH_SCALE ^ (generation + 3)) / (1 - LENGTH_SC
DrawDendrimer generation, picDendrimer.ScaleWidth / 2, picDendrimer.ScaleHeight - 2500, begin_length1, -PI

'This creates the bottom/ right portion of the dendrimer.
begin_length2 = (picDendrimer.ScaleHeight - 100) / ((1 - LENGTH_SCALE ^ (generation + 1)) / (1 - LENGTH_SC
DrawDendrimer generation, picDendrimer.ScaleWidth / 2, picDendrimer.ScaleHeight - 2500, begin_length2, PI

'This creates the bottom/ left portion of the dendrimer.
begin_length3 = (picDendrimer.ScaleHeight - 100) / ((1 - LENGTH_SCALE ^ (generation + 1)) / (1 - LENGTH_SC
DrawDendrimer generation, picDendrimer.ScaleWidth / 2, picDendrimer.ScaleHeight - 2500, begin_length3, PI

'This creates the right portion of the dendrimer.
begin_length4 = (picDendrimer.ScaleHeight - 100) / ((1 - LENGTH_SCALE ^ (generation + 1)) / (1 - LENGTH_SC
DrawDendrimer generation, picDendrimer.ScaleWidth / 2, picDendrimer.ScaleHeight - 2500, begin_length4, PI

End Sub
```

Dendrimer_Growth2:

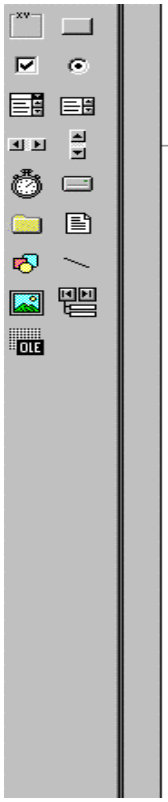


```
Timer1                                                    Timer

Private Sub Timer1_Timer()
Static indicator As Integer
     'The indicator determines the generation that will be displayed.
    DoEvents
    If indicator = 0 Then
    imgVisible(0).Picture = Image0(0).Picture

    ElseIf indicator = 1 Then
    imgVisible(0).Picture = Image1(1).Picture

    ElseIf indicator = 2 Then
    imgVisible(0).Picture = Image2(1).Picture

    ElseIf indicator = 3 Then
    imgVisible(0).Picture = Image3(1).Picture

    ElseIf indicator = 4 Then
    imgVisible(0).Picture = Image4(1).Picture

    ElseIf indicator = 5 Then
    imgVisible(0).Picture = Image5(1).Picture

    ElseIf indicator = 6 Then
    imgVisible(0).Picture = Image6(1).Picture

    ElseIf indicator = 7 Then
    imgVisible(0).Picture = Image7(1).Picture
    End If

    ' When the indicator reaches seven it will reset the program
    ' and the program will continue indefinitely.

    indicator = indicator + 1
    If indicator > 7 Then indicator = 0

End Sub
```

Dendrimer_Growth3:

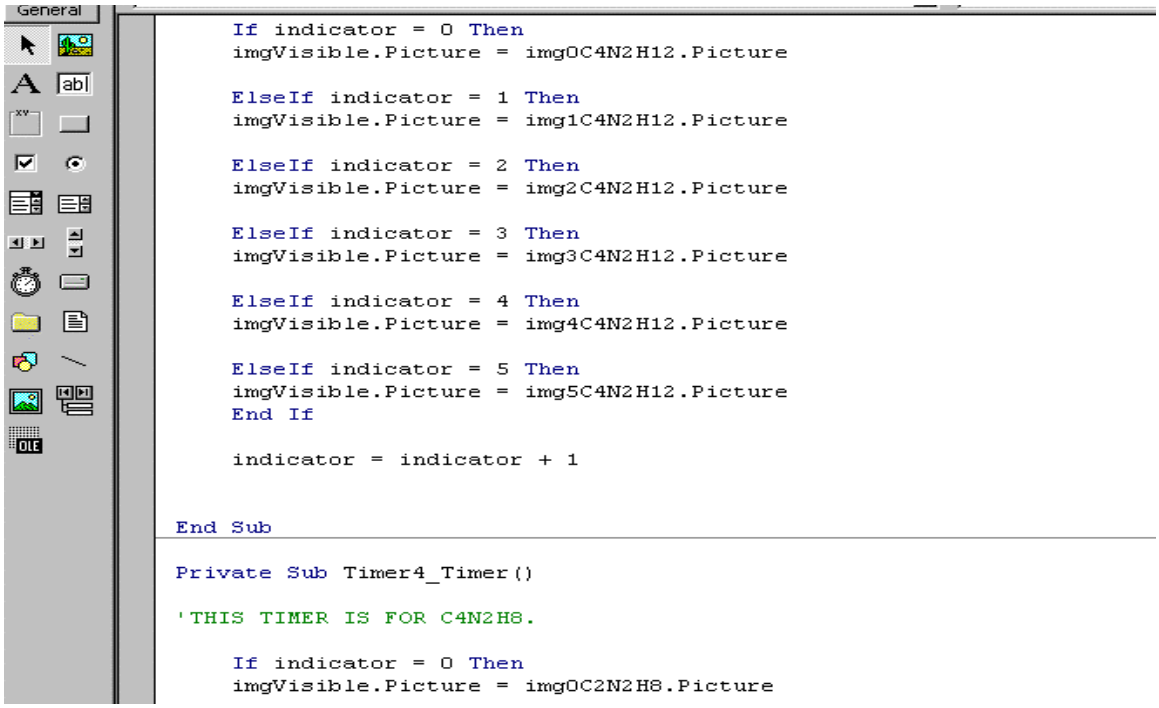


```
Option Explicit

Public indicator As Integer

Private Sub cmbBox_Click()
     If cmbBox.Text = "NH3" Then
     Call Timer1_Timer
     Timer1.Enabled = True
     Timer2.Enabled = False
     Timer3.Enabled = False
     Timer4.Enabled = False

     indicator = 0

     ElseIf cmbBox.Text = "N2H4" Then
     Call Timer2_Timer
     Timer1.Enabled = False
     Timer2.Enabled = True
     Timer3.Enabled = False
     Timer4.Enabled = False

     indicator = 0

     ElseIf cmbBox.Text = "C4N2H12" Then
     Call Timer3_Timer
     Timer1.Enabled = False
     Timer2.Enabled = False
     Timer3.Enabled = True
     Timer4.Enabled = False

     indicator = 0

     ElseIf cmbBox.Text = "C2N2H8" Then
     Call Timer4_Timer
     Timer1.Enabled = False
     Timer2.Enabled = False
     Timer3.Enabled = False
     Timer4.Enabled = True
```

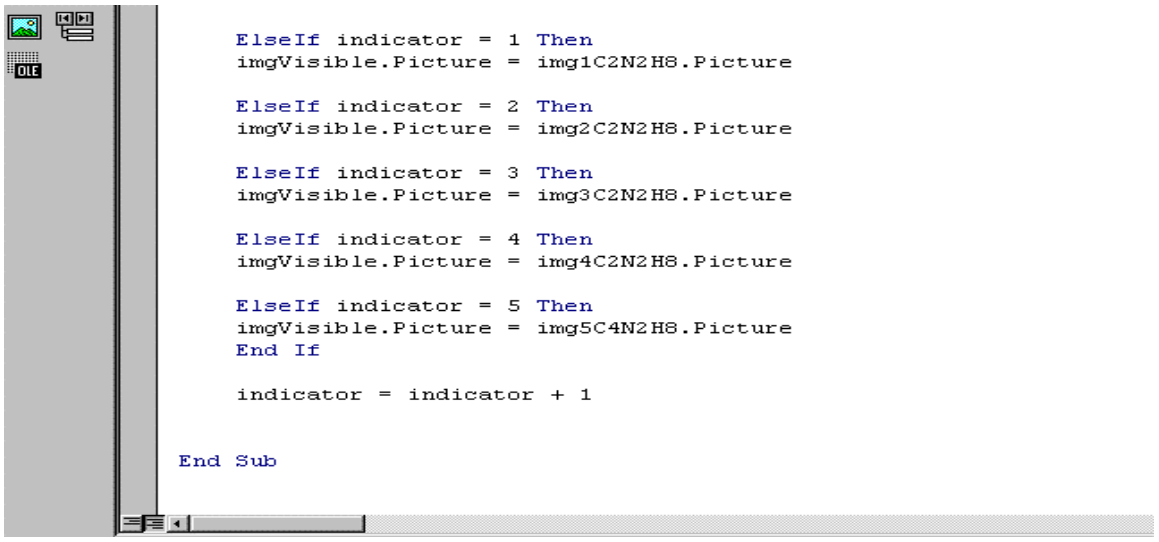


```
Private Sub Form_Load()
    cmbBox.AddItem "NH3"
    cmbBox.AddItem "N2H4"
    cmbBox.AddItem "C4N2H12"
    cmbBox.AddItem "C2N2H8"
End Sub

Private Sub Timer1_Timer()

'THIS TIMER IS FOR NH3.

    If indicator = 0 Then
    imgVisible.Picture = imgONH3.Picture

    ElseIf indicator = 1 Then
    imgVisible.Picture = img1NH3.Picture

    ElseIf indicator = 2 Then
    imgVisible.Picture = img2NH3.Picture

    ElseIf indicator = 3 Then
    imgVisible.Picture = img3NH3.Picture

    ElseIf indicator = 4 Then
    imgVisible.Picture = img4NH3.Picture

    ElseIf indicator = 5 Then
    imgVisible.Picture = img5NH3.Picture

    ElseIf indicator = 6 Then
    imgVisible.Picture = img6NH3.Picture
    End If

    indicator = indicator + 1
```

```
End Sub

Private Sub Timer2_Timer()

'THIS TIMER IS FOR N2H4.

    If indicator = 0 Then
    imgVisible.Picture = imgON2H4.Picture

    ElseIf indicator = 1 Then
    imgVisible.Picture = img1N2H4.Picture

    ElseIf indicator = 2 Then
    imgVisible.Picture = img2N2H4.Picture

    ElseIf indicator = 3 Then
    imgVisible.Picture = img3N2H4.Picture

    ElseIf indicator = 4 Then
    imgVisible.Picture = img4N2H4.Picture

    ElseIf indicator = 5 Then
    imgVisible.Picture = img5N2H4.Picture

    ElseIf indicator = 6 Then
    imgVisible.Picture = img6N2H4.Picture
    End If

    indicator = indicator + 1


End Sub

Private Sub Timer3_Timer()

'THIS TIMER IS FOR C4N2H12.
```



```
    If indicator = 0 Then
    imgVisible.Picture = imgOC4N2H12.Picture

    ElseIf indicator = 1 Then
    imgVisible.Picture = img1C4N2H12.Picture

    ElseIf indicator = 2 Then
    imgVisible.Picture = img2C4N2H12.Picture

    ElseIf indicator = 3 Then
    imgVisible.Picture = img3C4N2H12.Picture

    ElseIf indicator = 4 Then
    imgVisible.Picture = img4C4N2H12.Picture

    ElseIf indicator = 5 Then
    imgVisible.Picture = img5C4N2H12.Picture
    End If

    indicator = indicator + 1


End Sub

Private Sub Timer4_Timer()

'THIS TIMER IS FOR C4N2H8.

    If indicator = 0 Then
    imgVisible.Picture = imgOC2N2H8.Picture
```



```
    ElseIf indicator = 1 Then
    imgVisible.Picture = img1C2N2H8.Picture

    ElseIf indicator = 2 Then
    imgVisible.Picture = img2C2N2H8.Picture

    ElseIf indicator = 3 Then
    imgVisible.Picture = img3C2N2H8.Picture

    ElseIf indicator = 4 Then
    imgVisible.Picture = img4C2N2H8.Picture

    ElseIf indicator = 5 Then
    imgVisible.Picture = img5C4N2H8.Picture
    End If

    indicator = indicator + 1


End Sub
```