Pentium Paging Architecture

Submitted To – Mr. Govardhan Hegde

Team members:

Registration number	Name	Roll no
150905138	Aman Singh Thakur	33
150905352	Deepti Narayan	60
150905302	Shakha Mishra	55
150905120	Shashwat Dixit	27
150905074	Snigdha Shekhar	16

Introduction

Paging is a memory management scheme that permits the physical address space of a process to be non-contiguous. In this, the memory is divided into fixed size pages which are mapped to frames in the physical memory. Paging avoids external fragmentation and the need for compaction, however internal fragmentation is still there. It also solves the considerable problem of fitting memory chunks of varying sizes onto the backing store; most memory management schemes used before the introduction of paging suffered from this problem. The problem arises because, when some code fragments or data residing in main memory need to be swapped out, space must be found on the backing store. It allows faster access to the memory. Because of its advantages over earlier methods, paging in its various forms is used in most operating systems.

Methodology

The implementation of paging involves breaking physical memory into fixed-sized blocks called frames and breaking logical memory into blocks of the same size called pages. When a process is to be executed, its pages are loaded into any available memory frames from their source (a file system or the backing store). The backing store is divided into fixed-sized blocks that are of the same size as the memory frames. The hardware support for paging is shown in figure 1. Every address generated by the CPU is divided into two parts: a page number (p) and a page offset (d). The page number is used as an index into a page table. The page table contains the base address of each page in physical memory. This base address is combined with the page offset to define the physical memory address that is sent to the memory unit. The paging model of memory is shown in figure 2.

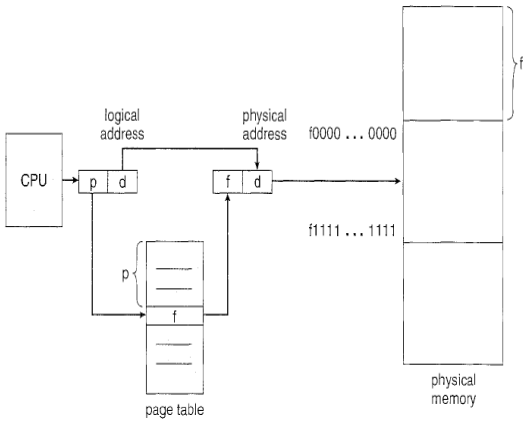


Figure 1

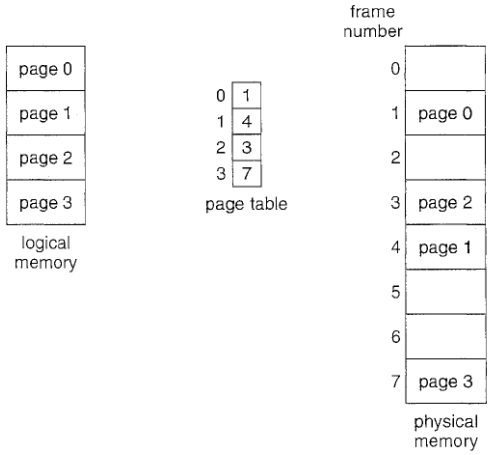
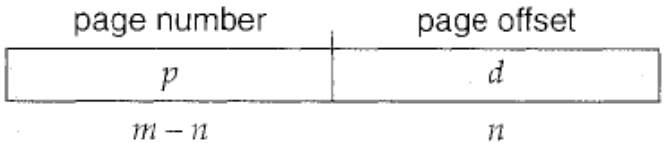


Figure 2

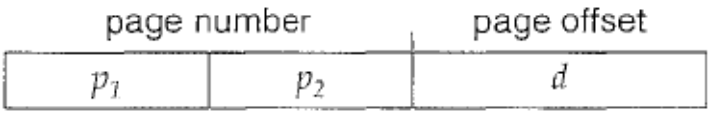
The page size (like the frame size) is defined by the hardware. If the size of the logical address space is 2m, and a page size is 2n, then the high-order m- n bits of a logical address designate the page number, and the n low-order bits designate the page offset. Thus, the logical address is as follows:



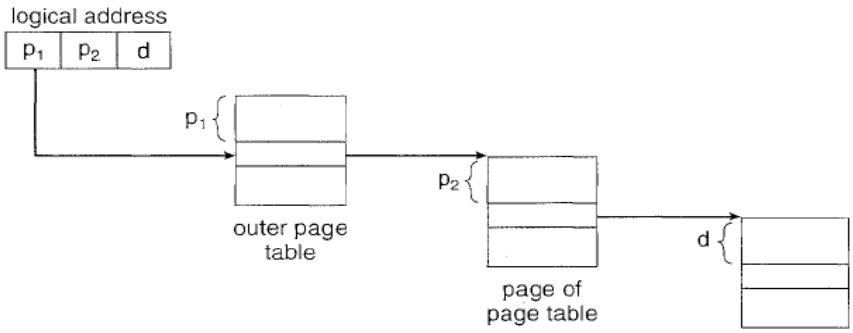
where p is an index into the page table and d is the displacement within the page.

Hierarchical Paging

Most modern computer systems support a large logical address space (232 to 264). In such an environment, the page table itself becomes excessively large. One simple solution to this problem is to divide the page table into smaller pieces. We can accomplish this division in several ways. One way is to use a two-level paging algorithm, in which the page table itself is also paged.

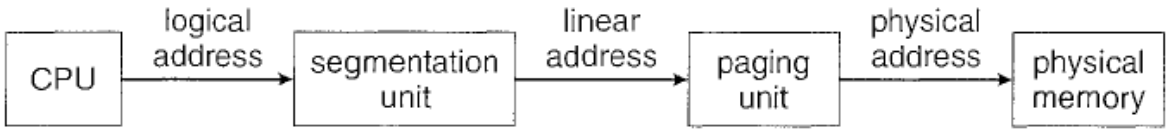


where p1 is an index into the outer page table and p2 is the displacement within the page of the outer page table. This scheme is also known as a forward mapped page table.

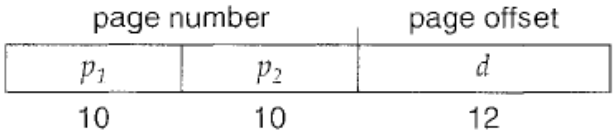


Pentium Paging

In Pentium systems, the CPU generates logical addresses, which are given to the segmentation unit. The segmentation unit produces a linear address for each logical address. The linear address is then given to the paging unit, which in turn generates the physical address in main memory. Thus, the segmentation and paging units form the equivalent of the memory-management unit (MMU). This scheme is shown in the figure below.



The Pentium architecture allows a page size of either 4 KB or 4 MB. For 4-KB pages, the Pentium uses a two-level paging scheme in which the division of the 32-bit linear address is as follows:



The 10 high-order bits reference an entry in the outermost page table, which the Pentium terms the page directory. The CR3 register points to the page directory for the current process. The page directory entry points to an inner page table that is indexed by the contents of the innermost 10 bits in the linear address. Finally, the low-order bits 0-11 refer to the offset in the 4-KB page pointed to in the page table. One entry in the page directory is the Page Size flag, which-if set, indicates that the size of the page frame is 4 MB and not the standard 4 KB. If this flag is set, the page directory points directly to the 4-MB page frame, bypassing the inner page table; and the 22 low-order bits in the linear address refer to the offset in the 4-MB page frame.

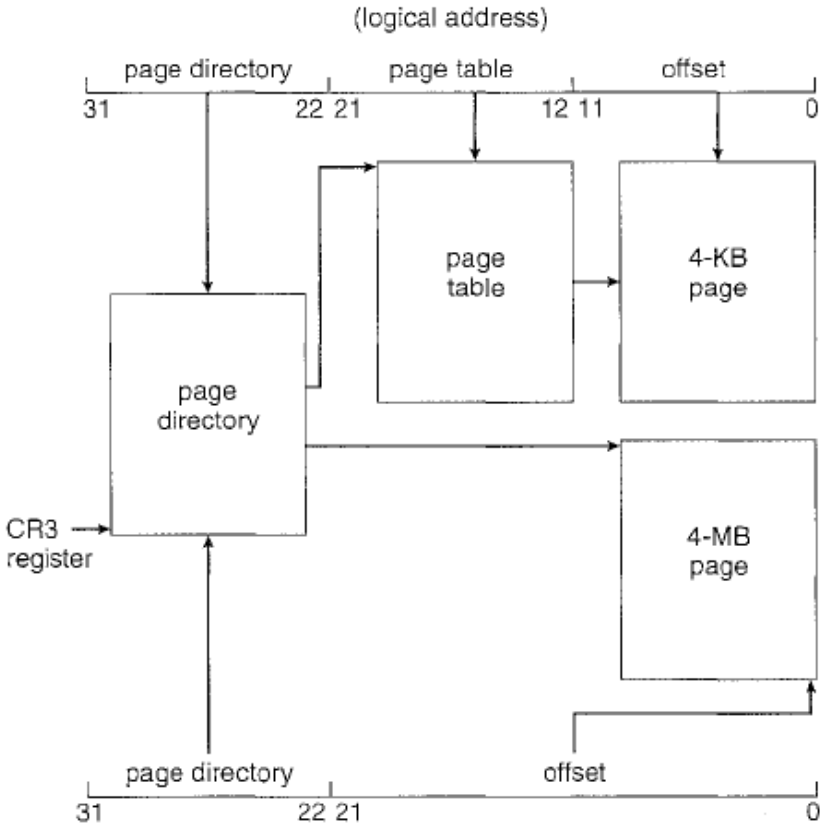


Figure : Pentium Paging Architecture

Implementation

To demonstrate the working of the Pentium paging architecture, we have created a software on JAVA Swings on NETBEANS IDE. The frontend is handled by AWT libraries such as JFrame and the backend is handled by JAVA. This GUI based interactive software will help users to understand the Pentium paging architecture with much more understanding.

We have taken as input a 32 bit logical address in the file 'inputPaging.java'. Along with that, we have used a toggle button for the Page Size Flag. By default, the Page Size flag is not set. On clicking the Find Page button, depending on the value of page size flag, we get new window pop ups with 4KB or 4MB page table and offset values.

In 'Pagingfourkb.java', the logical address is broken down into Page directory (10 bits), Page table (10 bits) and offset (12 bits) values using substring function in java. Each value is then converted into decimal using predefined functions in Integer class. Similarly, in 'Pagingfourmb.java', the logical address is broken down into Page Table (10 bits) and offset (22 bits). On clicking 'Find physical address' button, new window pops up in which framing is done.
It's impossible to store all frame values to corresponding page values. Hence, we find the exact page number using page table in 4MB pages or two-level paging schema in 4KB page. For 4KB page, their exists 220 frames. Hence, we generate a random value between 0-(220-1). We then append the offset to that value and get the 32 bit linear address in physical memory. For 4MB page, their exists 210 frames in main memory. Hence, we generate a random number between 0-(210-1) and the same procedure is applied.

The 32 bit physical address is converted into decimal and hexadecimal values using predefined function 'toString()'.

Code

InputPaging.java

```
package my;

public class InputPaging extends javax.swing.JFrame {
   int PageSizeFlag=0;

   public InputPaging() {
      initComponents();
      jLabel3.setVisible(false);
   }


   @SuppressWarnings("unchecked")
   // <editor-fold defaultstate="collapsed" desc="Generated Code">
   private void initComponents() {
```

```
        jButton1 = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jToggleButton1 = new javax.swing.JToggleButton();
        jLabel3 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jButton1.setText("Find Page");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        jLabel1.setText("Enter 32 bit logical Address");

        jLabel2.setText("PENTIUM  PAGING  TECHNIQUE");

        jToggleButton1.setText("Set Page Size Flag");
        jToggleButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jToggleButton1ActionPerformed(evt);
            }
        });

        jLabel3.setForeground(new java.awt.Color(255, 0, 0));
        jLabel3.setText("Error");

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel3)
                .addGap(179, 179, 179))
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jToggleButton1)
                .addGap(118, 118, 118))
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
                .addContainerGap(18, Short.MAX_VALUE)
                .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, 355,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
            .addGap(0, 27, Short.MAX_VALUE))
         .addGroup(layout.createSequentialGroup()
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
               .addComponent(jLabel2)
               .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                  .addComponent(jLabel1)
                  .addGap(9, 9, 9)))
            .addGap(97, 97, 97))
         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jButton1)
            .addGap(141, 141, 141))
      );
      layout.setVerticalGroup(
         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
         .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
            .addGap(17, 17, 17)
            .addComponent(jLabel2)
            .addGap(18, 18, 18)
            .addComponent(jLabel1)
            .addGap(53, 53, 53)
            .addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(29, 29, 29)
            .addComponent(jToggleButton1)
            .addGap(37, 37, 37)
            .addComponent(jButton1)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 15,
Short.MAX_VALUE)
            .addComponent(jLabel3)
            .addGap(22, 22, 22))
      );

      pack();
   }// </editor-fold>

   private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
      String logicaladdress = jTextField1.getText();
      if (logicaladdress.length()!=32){
         jLabel3.setText("Logical address has only "+logicaladdress.length()+" bits");
         jLabel3.setVisible(true);
      }
      else{
         jLabel3.setVisible(false);
         if(PageSizeFlag==0)
```

```
        {
            Pagingfourkb p = new Pagingfourkb(jTextField1.getText());
            p.setVisible(true);
        }else{
            Pagingfourmb p = new Pagingfourmb(jTextField1.getText());
            p.setVisible(true);
        }
    }
}

private void jToggleButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    PageSizeFlag=1;
    jToggleButton1.setText("Page Size Flag is now Set");
}

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new InputPaging().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JTextField jTextField1;
private javax.swing.JToggleButton jToggleButton1;
// End of variables declaration
}
```

Pagingfourkb.java

```
package my;

public class Pagingfourkb extends javax.swing.JFrame {

    static String logicaladdress;
    static String PageDirectory, PageTable, fourkbpage;

    public Pagingfourkb() {
        initComponents();
    }

    Pagingfourkb(String logicaladdress) {
```

```
     this.logicaladdress=logicaladdress;
    initComponents();
    fourkbpage=logicaladdress.substring(20,32);
    PageTable=logicaladdress.substring(10,20);
    PageDirectory=logicaladdress.substring(0,10);
    jLabel4.setText(PageDirectory);
    jLabel9.setText(PageTable);
    jLabel15.setText(fourkbpage);
    jLabel6.setText(String.valueOf(Integer.parseInt(PageDirectory,2)));
    jLabel11.setText(String.valueOf(Integer.parseInt(PageTable,2)));
    jLabel16.setText(String.valueOf(Integer.parseInt(fourkbpage,2)));
}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jLabel6 = new javax.swing.JLabel();
    jLabel7 = new javax.swing.JLabel();
    jLabel8 = new javax.swing.JLabel();
    jLabel9 = new javax.swing.JLabel();
    jLabel10 = new javax.swing.JLabel();
    jLabel11 = new javax.swing.JLabel();
    jLabel12 = new javax.swing.JLabel();
    jLabel13 = new javax.swing.JLabel();
    jLabel14 = new javax.swing.JLabel();
    jLabel15 = new javax.swing.JLabel();
    jLabel16 = new javax.swing.JLabel();
    jButton1 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("Paging using 4KB Pages");
    jLabel2.setText("Page Directory");
    jLabel3.setText("Binary Value : ");
    jLabel4.setText("Insert Binary Value");
    jLabel5.setText("Decimal Value :");
    jLabel6.setText("Insert Decimal Value");
    jLabel7.setText("Page Table");
    jLabel8.setText("Binary Value :");
    jLabel9.setText("Insert Binary Value");
    jLabel10.setText("Decimal Value :");
```

```
        jLabel11.setText("Insert Decimal Value");
        jLabel12.setText("4-KB Page ");
        jLabel13.setText("Binary Value :");
        jLabel14.setText("Decimal Value");
        jLabel15.setText("Insert Binary Value");
        jLabel16.setText("Insert Decimal Value");

        jButton1.setText("Find Physical Address");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(layout.createSequentialGroup()
                        .addGap(149, 149, 149)
                        .addComponent(jLabel2))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(79, 79, 79)
                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addComponent(jLabel3)
                            .addComponent(jLabel5)
                            .addComponent(jLabel8)
                            .addComponent(jLabel10))
                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addGroup(layout.createSequentialGroup()
                                .addGap(41, 41, 41)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                    .addComponent(jLabel6)
                                    .addComponent(jLabel4)))
                            .addGroup(layout.createSequentialGroup()
                                .addGap(45, 45, 45)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                    .addComponent(jLabel11)
                                    .addComponent(jLabel9)))))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(135, 135, 135)
                        .addComponent(jLabel1))
                    .addGroup(layout.createSequentialGroup()
```

```
                .addGap(164, 164, 164)
                .addComponent(jLabel7))
            .addGroup(layout.createSequentialGroup()
                .addGap(170, 170, 170)
                .addComponent(jLabel12))
            .addGroup(layout.createSequentialGroup()
                .addGap(88, 88, 88)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jLabel13)
                    .addComponent(jLabel14))
                .addGap(42, 42, 42)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jLabel16)
                    .addComponent(jLabel15)))
            .addGroup(layout.createSequentialGroup()
                .addGap(109, 109, 109)
                .addComponent(jButton1)))
        .addContainerGap(61, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(17, 17, 17)
            .addComponent(jLabel1)
            .addGap(18, 18, 18)
            .addComponent(jLabel2)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(28, 28, 28)
                    .addComponent(jLabel4))
                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .addComponent(jLabel3)))
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel5)
                .addComponent(jLabel6))
            .addGap(18, 18, 18)
            .addComponent(jLabel7)
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel8)
                .addComponent(jLabel9))
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel10)
```

```
            .addComponent(jLabel11))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(jLabel12)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel13)
            .addComponent(jLabel15))
        .addGap(18, 18, 18)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel14)
            .addComponent(jLabel16))
        .addGap(18, 18, 18)
        .addComponent(jButton1)
        .addContainerGap(20, Short.MAX_VALUE))
    );

    pack();
}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

    Framingfourkb f = new Framingfourkb(logicaladdress);
    f.setVisible(true);
}

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Pagingfourkb(logicaladdress).setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel15;
private javax.swing.JLabel jLabel16;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
```

```
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    // End of variables declaration
}
```

Framingfourfb.java

```
package my;

import java.math.BigInteger;
import java.util.Random;

public class Framingfourkb extends javax.swing.JFrame {

    static String logicaladdress;
    static String PageDirectory, PageTable, fourkbpage;
    static BigInteger frame;
    public Framingfourkb() {
        initComponents();
    }

    Framingfourkb(String logicaladdress) {
        initComponents();
        fourkbpage=logicaladdress.substring(20,32);
        PageTable=logicaladdress.substring(10,20);
        PageDirectory=logicaladdress.substring(0,10);
        Random rand = new Random();
        int n = rand.nextInt(1048576) + 0;
        jLabel10.setText(String.valueOf(n));
        String physicaladdress = Integer.toBinaryString(n)+fourkbpage;
        jLabel12.setText(String.valueOf(n)+" + "+String.valueOf(Integer.parseInt(fourkbpage,2))+" =
"+physicaladdress);
        jLabel7.setText(String.valueOf(Integer.parseInt(physicaladdress,2)));
        int physical = Integer.parseInt(physicaladdress,2);
        jLabel9.setText("0x"+Integer.toString(physical,16));
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jRadioButton1 = new javax.swing.JRadioButton();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
```

```
jLabel5 = new javax.swing.JLabel();
jLabel6 = new javax.swing.JLabel();
jLabel7 = new javax.swing.JLabel();
jLabel8 = new javax.swing.JLabel();
jLabel9 = new javax.swing.JLabel();
jButton1 = new javax.swing.JButton();
jLabel10 = new javax.swing.JLabel();
jLabel11 = new javax.swing.JLabel();
jLabel12 = new javax.swing.JLabel();

jRadioButton1.setText("jRadioButton1");

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

jLabel1.setText("Physical Address");
jLabel2.setText("Frame Number to the generated 4 KB Page Number");
jLabel4.setText("Frame No :");
jLabel5.setText("Calculating result by Frame No + Offset (append)");
jLabel6.setText("Physical Address (in Decimal) ");
jLabel7.setText("Insert Physical Address");
jLabel8.setText("Physical Address (in Hexadecimal)");
jLabel9.setText("Insert Physical Address");
jButton1.setText("Exit");

jButton1.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      jButton1ActionPerformed(evt);
   }
});

jLabel10.setText("Insert Frame No");
jLabel11.setText("Physical Address (in Binary) ");
jLabel12.setText("Insert Physical Address");

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
   layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
   .addGroup(layout.createSequentialGroup()
      .addGap(105, 105, 105)
      .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.CENTER)
         .addComponent(jLabel8)
         .addComponent(jLabel9)
         .addComponent(jButton1)
         .addComponent(jLabel2)
         .addGroup(layout.createSequentialGroup()
            .addComponent(jLabel4)
```

```
                .addGap(82, 82, 82)
                .addComponent(jLabel10))
            .addComponent(jLabel5)
            .addComponent(jLabel6)
            .addComponent(jLabel7)
            .addComponent(jLabel1)
            .addComponent(jLabel12)
            .addComponent(jLabel11))
        .addContainerGap(119, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(18, 18, 18)
            .addComponent(jLabel1)
            .addGap(18, 18, 18)
            .addComponent(jLabel2)
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel4)
                .addComponent(jLabel10))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jLabel5)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jLabel11)
            .addGap(18, 18, 18)
            .addComponent(jLabel12)
            .addGap(18, 18, 18)
            .addComponent(jLabel6)
            .addGap(18, 18, 18)
            .addComponent(jLabel7)
            .addGap(18, 18, 18)
            .addComponent(jLabel8)
            .addGap(18, 18, 18)
            .addComponent(jLabel9)
            .addGap(18, 18, 18)
            .addComponent(jButton1)
            .addContainerGap(23, Short.MAX_VALUE))
    );

    pack();
}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    System.exit(0);
}
```

```
    public static void main(String args[]) {

        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Framingfourkb(logicaladdress).setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel10;
    private javax.swing.JLabel jLabel11;
    private javax.swing.JLabel jLabel12;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    private javax.swing.JRadioButton jRadioButton1;
    // End of variables declaration
}
```

Framingfourmb.java

```
package my;

import java.math.BigInteger;
import java.util.Random;

public class Framingfourmb extends javax.swing.JFrame {
    static String logicaladdress;
    static String PageDirectory, fourmbpage;
    static BigInteger frame;

    public Framingfourmb() {
        initComponents();
    }

     Framingfourmb(String logicaladdress) {
        initComponents();
        fourmbpage=logicaladdress.substring(10,32);
        PageDirectory=logicaladdress.substring(0,10);
```

```
      //frame = BigInteger.valueOf(Integer.parseInt(PageDirectory,2)-
1).multiply(BigInteger.valueOf(4194304));
      Random rand = new Random();
      int n = rand.nextInt(1024) + 0;
      jLabel5.setText(String.valueOf(n));
      String physicaladdress = Integer.toBinaryString(n)+fourmbpage;
      jLabel13.setText(String.valueOf(n)+" + "+String.valueOf(Integer.parseInt(fourmbpage,2))+" =
"+physicaladdress);
      //jLabel12.setText(String.valueOf(n)+" + "+String.valueOf(Integer.parseInt(fourmbpage,2))+" =
"+physicaladdress);
      jLabel7.setText(String.valueOf(Integer.parseInt(physicaladdress,2)));
      int physical = Integer.parseInt(physicaladdress,2);
      jLabel10.setText("0x"+Integer.toString(physical,16));
   }

   @SuppressWarnings("unchecked")
   // <editor-fold defaultstate="collapsed" desc="Generated Code">
   private void initComponents() {

      jLabel11 = new javax.swing.JLabel();
      jLabel1 = new javax.swing.JLabel();
      jLabel2 = new javax.swing.JLabel();
      jLabel4 = new javax.swing.JLabel();
      jLabel5 = new javax.swing.JLabel();
      jLabel6 = new javax.swing.JLabel();
      jLabel7 = new javax.swing.JLabel();
      jLabel8 = new javax.swing.JLabel();
      jLabel9 = new javax.swing.JLabel();
      jLabel10 = new javax.swing.JLabel();
      jButton1 = new javax.swing.JButton();
      jLabel12 = new javax.swing.JLabel();
      jLabel13 = new javax.swing.JLabel();

      jLabel11.setText("jLabel11");

      setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
      jLabel1.setText("Physical Address");
      jLabel2.setText("Frame Number to the generated 4 MB Page Number");
      jLabel4.setText("Frame No :");
      jLabel5.setText("Insert Frame No");
      jLabel6.setText("Calculating Result by adding Frame No + Offset (offset)");
      jLabel7.setText("Physical Address (in Decimal)");
      jLabel8.setText("Insert Physical Address");
      jLabel9.setText("Physical Address (in Hexadecimal)");
      jLabel10.setText("Insert Physical Address");

      jButton1.setText("Exit");
```

```
jButton1.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      jButton1ActionPerformed(evt);
   }
});

jLabel12.setText("Physical Address (in Binary)");

jLabel13.setText("Insert Physical Address");

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
   layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
   .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
      .addContainerGap(67, Short.MAX_VALUE)
      .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.CENTER)
         .addComponent(jLabel9)
         .addComponent(jLabel10)
         .addComponent(jButton1)
         .addComponent(jLabel1)
         .addComponent(jLabel2)
         .addGroup(layout.createSequentialGroup()
            .addComponent(jLabel4)
            .addGap(128, 128, 128)
            .addComponent(jLabel5))
         .addComponent(jLabel6)
         .addComponent(jLabel7)
         .addComponent(jLabel8)
         .addComponent(jLabel13)
         .addComponent(jLabel12))
      .addGap(81, 81, 81))
);
layout.setVerticalGroup(
   layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
   .addGroup(layout.createSequentialGroup()
      .addGap(25, 25, 25)
      .addComponent(jLabel1)
      .addGap(18, 18, 18)
      .addComponent(jLabel2)
      .addGap(30, 30, 30)
      .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
         .addComponent(jLabel4)
         .addComponent(jLabel5))
      .addGap(32, 32, 32)
      .addComponent(jLabel6)
      .addGap(18, 18, 18)
```

```
            .addComponent(jLabel12)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 18,
Short.MAX_VALUE)
            .addComponent(jLabel13)
            .addGap(18, 18, 18)
            .addComponent(jLabel7)
            .addGap(18, 18, 18)
            .addComponent(jLabel8)
            .addGap(18, 18, 18)
            .addComponent(jLabel9)
            .addGap(18, 18, 18)
            .addComponent(jLabel10)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jButton1)
            .addGap(15, 15, 15))
        );

        pack();
    }// </editor-fold>

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

        System.exit(0);
    }

    public static void main(String args[]) {

        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Framingfourmb(logicaladdress).setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel10;
    private javax.swing.JLabel jLabel11;
    private javax.swing.JLabel jLabel12;
    private javax.swing.JLabel jLabel13;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
```

```
    private javax.swing.JLabel jLabel9;
    // End of variables declaration
}
```

Conclusion

Using Pentium paging architecture, we have demonstrated that it's possible to have multiple page sizes. For heavy software's like PC games, larger page sizes are necessary as the initial file size is also large and there are a lot of dynamic factors. This is not followed for smaller software's because speed and accuracy is more important and memory used is very less. This project also demonstrates page tables and two-level page schema. Real Pentium use Segmentation and Paging together to give faster performance with segments as large as 4GB. To improve efficiency of physical memory, Intel Pentium page tables can be swapped to disk. In this case, An invalid bit is used in the page directory entry to indicate whether the table to which the entry is pointing in memory or on disk.

Screenshots

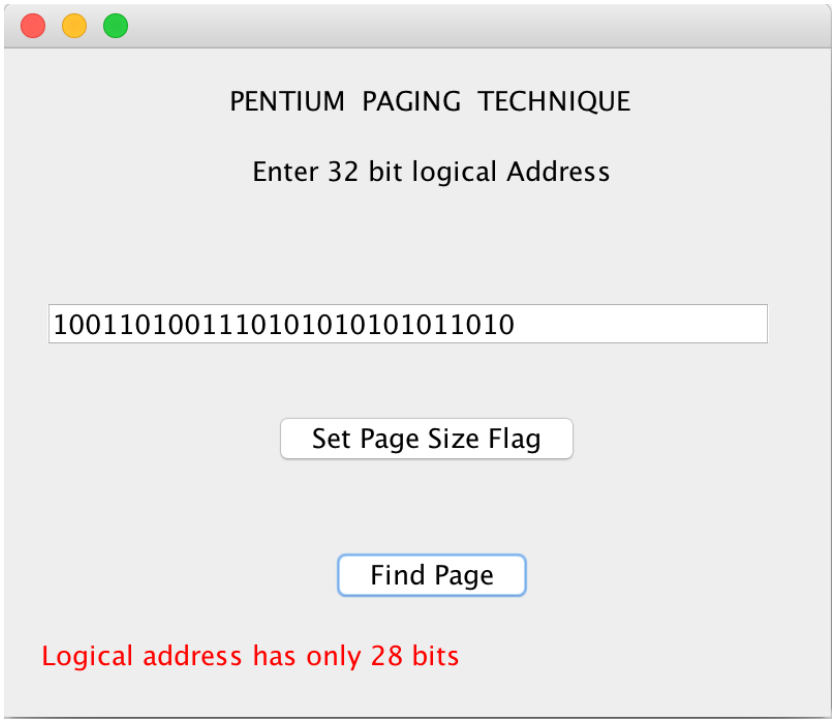


Figure : Error Handling

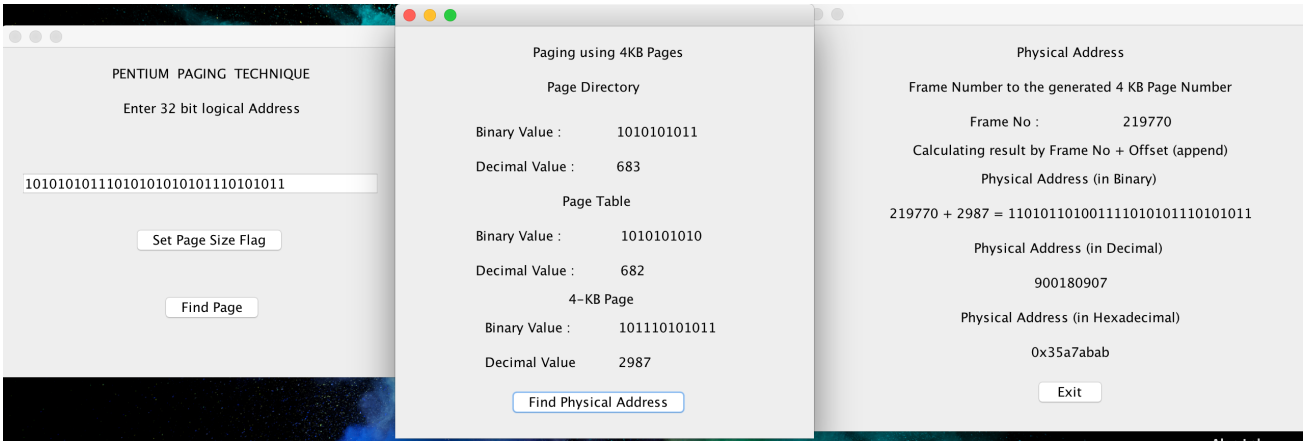


Figure : 4KB Page

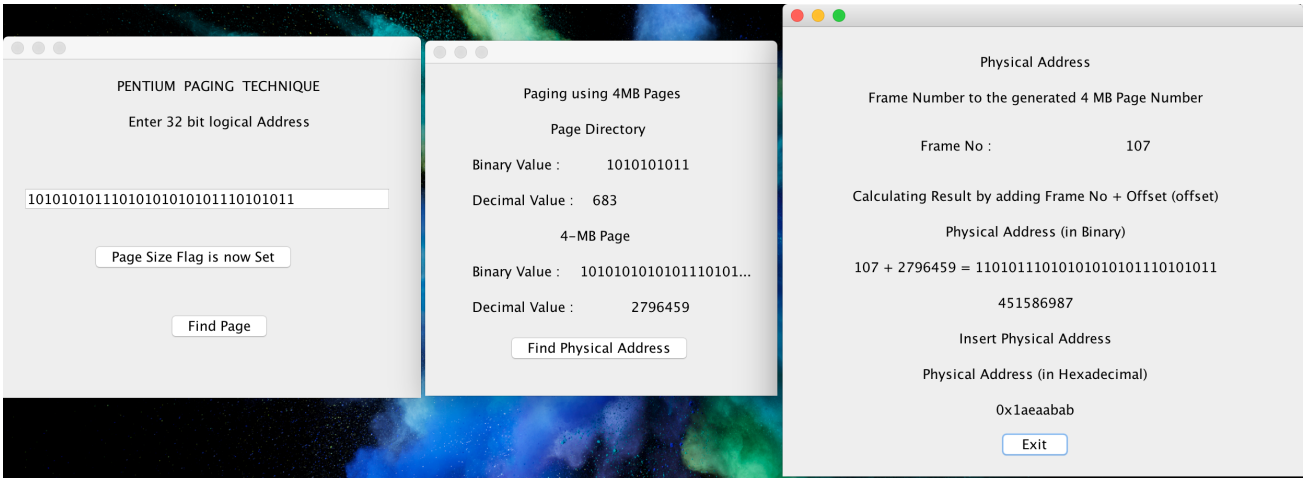


Figure : 4MB Page