Project: Computer Systems

STATUS:

This project is complete. I created a kitchen scene using ASM and have it in an axonometric view. This is at a position 45 degrees from vertical or horizontal. The kitchen features a table, chair, counter top with cupboards, a sink and a window with a blind.

There are on click events on the chair, sink and window, where the chair will move back from the table, the sink will fill up and the blind will pull down on the window. There is also a 30x30 area in the top right, which will let one exit the program safely.

Writing the program:

The program starts with DRAWSCENE, DRAWXBUTTON, DRAWFRAME and MOUSEACTION respectively. The DRAWSCENE draws the general kitchen scene as outlined below. DRAWXBUTTON has a FILLBOX and x drawn inside it, this is also the area to click to safely exit the program. DRAWFRAME draws a one pixel thick line around the edge of the screen to frame the scene.

```
MAIN    PROC    FAR

        PUSH    DS                          ;These 5 lines are
        MOV     AX,0                        ;required for all .ASM
        PUSH    AX                          ;programs.
        MOV     AX,DSEG                     ;
        MOV     DS,AX                       ;



                                            ;The program begins here.
        MOV     AH,00H                      ;Video: 640x480
        MOV     AL,18
        INT     10H

        CALL    DRAWSCENE
        CALL    DRAWXBUTTON
        MOV     AL,0DH
        CALL    DRAWFRAME
        CALL    MOUSEACTION


        MOV     AH,00H
        MOV     AL,2            ;Reset the screen to normal
        INT     10H


        RET                                 ;The program ends here.

MAIN    ENDP
```

The MOUSEACTION controls all interaction with the scene. It starts by showing the mouse, and then looping to check if the mouse is clicked, when it is clicked it send is to the handler, ONCLICK.

```
;-------------------------------------------------------------
;       MOUSEACTION
;       Description:    Controls the mouse actions
;       Input params:   None
;       Reg Effected:   None
;-------------------------------------------------------------

MOUSEACTION PROC
        MOV     AX,01H                          ;Shows mouse cursor
        INT     33H

CHECKFORCLICK:

        MOV     AX,05H                          ;Checks for click
        MOV     BX,0H
        INT     33H
        CMP     BX,1
        JE      ACTION
        JMP     CHECKFORCLICK

ACTION:
        CALL    ONCLICK                         ;Performs click action
        CMP     CX,610                          ;Then checks for exit
        JB      CHECKFORCLICK
        CMP     DX,30
        JA      CHECKFORCLICK

        RET
MOUSEACTION ENDP
```

The ONCLICK function checks the coordinates that were clicked, and using these decides what action to take. This is the only function that effects the registers, as it decides whether or not there is a safe exit in the parent MOUSEACTION.

```
;-------------------------------------------------------------
;       ONCLICK
;       Description:    Handles mouse click
;       Input params:   None
;
;
;       Reg Effected:   CX - Column clicked
;                       DX - Row clicked
;-------------------------------------------------------------

ONCLICK PROC

OPTION1:


      CMP       CX,40                   ;Checks for window area click
      JB        ENDCLICK
      CMP       CX,180
      JA        OPTION2
      CMP       DX,130
      JB        OPTION2
      CMP       DX,370
      JA        OPTION2

      MOV       AX,00H                  ;Hides the cursor
      INT       33H                     ;Cancels overwritten colours


      MOV       AL,0FH
      CALL      DRAWBLINDS              ;Animates the blinds

      MOV       AX,01H                  ;Reshows the mouse cursor
      INT       33H
```

```
        MOV     AX,01H                  ;Reshows the mouse cursor
        INT     33H

        JMP     ENDCLICK

OPTION2:
        CMP     CX,340                  ;Checks if sink area clicked
        JB      OPTION3
        CMP     CX,430
        JA      ENDCLICK
        CMP     DX,215
        JB      ENDCLICK
        CMP     DX,305
        JA      OPTION3

        MOV     AX,00H
        INT     33H

        CALL    TAPANIM                 ;Animates filling sink
        MOV     AX,01H
        INT     33H

        JMP     ENDCLICK

OPTION3:
        CMP     CX,215                  ;Check if table/chair area
        JB      ENDCLICK                ;clicked
        CMP     CX,317
        JA      ENDCLICK
        CMP     DX,300
        JB      ENDCLICK
        CMP     DX,460
        JA      ENDCLICK

        MOV     AX,00H
        INT     33H

        MOV     AX,00H
        INT     33H

        CALL    MOVECHAIR               ;Animates the moving chair

        MOV     AX,01H
        INT     33H

ENDCLICK:

        RET
ONCLICK ENDP
```

To show an example of an animation, I will use the tap animation. This is when the sink is clicked, the tap turns on and the sink fills with water. When the sink is full enough, the tap turns off again. This is all done in the TAPANIM function.

```
;-------------------------------------------------------------
;       TAPANIM
;       Description:    Animation for filling the sink
;       Input params:   None
;       Reg Effected:   None
;-------------------------------------------------------------

TAPANIM PROC
        PUSH    CX
        PUSH    DX
        PUSH    AX
        PUSH    BX
        PUSH    SI
        PUSH    DI

        MOV     CX,412                          ;Draw stream from tap
        MOV     DX,258
        MOV     AL,01H
        MOV     SI,20
        CALL    DRAWVLINE

        MOV     CX,350                          ;Preparing sink fill
        MOV     DX,253                          ;loop
        MOV     DI,22
        MOV     SI,51
        MOV     BX,6

        MOV     BX,6
FILLING:
        CMP     BX,0                            ;Loop for filling
        JE      FINISHSINK
        DEC     BX
        DEC     CX
        DEC     DX
        INC     SI
        INC     DI
        CALL    DRAWDIAMOND
        CALL    DELAY
        JMP     FILLING

FINISHSINK:
        MOV     CX,412
        MOV     DX,257
        MOV     AL,07H
        MOV     SI,3
        CALL    DRAWVLINE                       ;Turn off the tap stream


        POP     DI
        POP     SI
        POP     BX
        POP     AX
        POP     DX
        POP     CX

        RET
TAPANIM ENDP
```

The tap animation starts with the tap turning on, with a blue DRAWVLINE, followed by the sink filling with a blue DRAWDIAMOND. The diamond expands by decreasing CX, and BX. It also increases the SI and DI values, so that it expands on all sides, not just moving up and to the left. When this loop is finished, it changes the DRAWVLINE, to a shorter SI value, and to a grey colour, to show the tap turning off. It is made shorter so that the water does not have a vertical grey line going through it.

```
;---------------------------------------------------------------
;       DRAWBLINDS
;       Description:    DRAWBLINDS
;       Input params:   None
;       Reg Effected:   None
;---------------------------------------------------------------

DRAWBLINDS PROC
        PUSH    CX
        PUSH    DX
        PUSH    AX
        PUSH    DI
        PUSH    SI

        MOV     DI,117
        MOV     SI,140
        MOV     CX,40
        MOV     DX,275                          ;Preparing blind animation


AGAINBLIND:
        CMP     DI,0
        JE      FINISHBLINDS
        DEC     DI
        CALL    DRAWUP45LINE
        INC     DX
        CALL    DELAY
        JMP     AGAINBLIND                      ;Pulling blinds and looping

FINISHBLINDS:
        POP     SI
        POP     DI
        POP     AX
        POP     DX
        POP     CX

        RET
DRAWBLINDS ENDP
```

To implement the blinds closing, I used the DRAWBLINDS function. In this, I find the location of the blinds, and then continue to draw continuous 45 degree lines with a delay, all in white. This gives the impression that the blinds are slowly closing.

```
;-------------------------------------------------------------
;       MOVECHAIR
;       Description:    Animation for moving the chair
;       Input params:   None
;       Reg Effected:   None
;-------------------------------------------------------------

MOVECHAIR PROC
        PUSH    AX
        PUSH    BX
        PUSH    CX
        PUSH    DX

        MOV     CX,285
        MOV     DX,360
        MOV     BX,0
        MOV     AL,03H
        CALL    DRAWCHAIR                       ;Blanks out the chair

CHAIRMOVELOOP:
        CMP     BX,20                           ;Chair moving loop
        JE      FINISHMOVECHAIR
        INC     BX

        MOV     AL,03H
        ADD     CX,BX
        SUB     DX,BX
        CALL    DRAWCHAIR                       ;Blanks chair

        MOV     AL,0FH
        CALL    DRAWTILES
        CALL    DRAWCOUNTERFRONT                ;Redrawing tiles and counter

        INC     CX
        DEC     DX
        MOV     AL,04H
        CALL    DRAWCHAIR                       ;Redrawing chair

        MOV     AL,05H
        MOV     CX,280
        MOV     DX,420
        CALL    DRAWTABLE                       ;Redrawing table
                                                ;over the chair
        MOV     CX,285
        MOV     DX,360
        JMP     CHAIRMOVELOOP                   ;Jumps back to start
                                                ;of the loop

FINISHMOVECHAIR:
```

```
FINISHMOVECHAIR:
        POP     DX
        POP     CX
        POP     BX
        POP     AX

        RET
MOVECHAIR ENDP
```

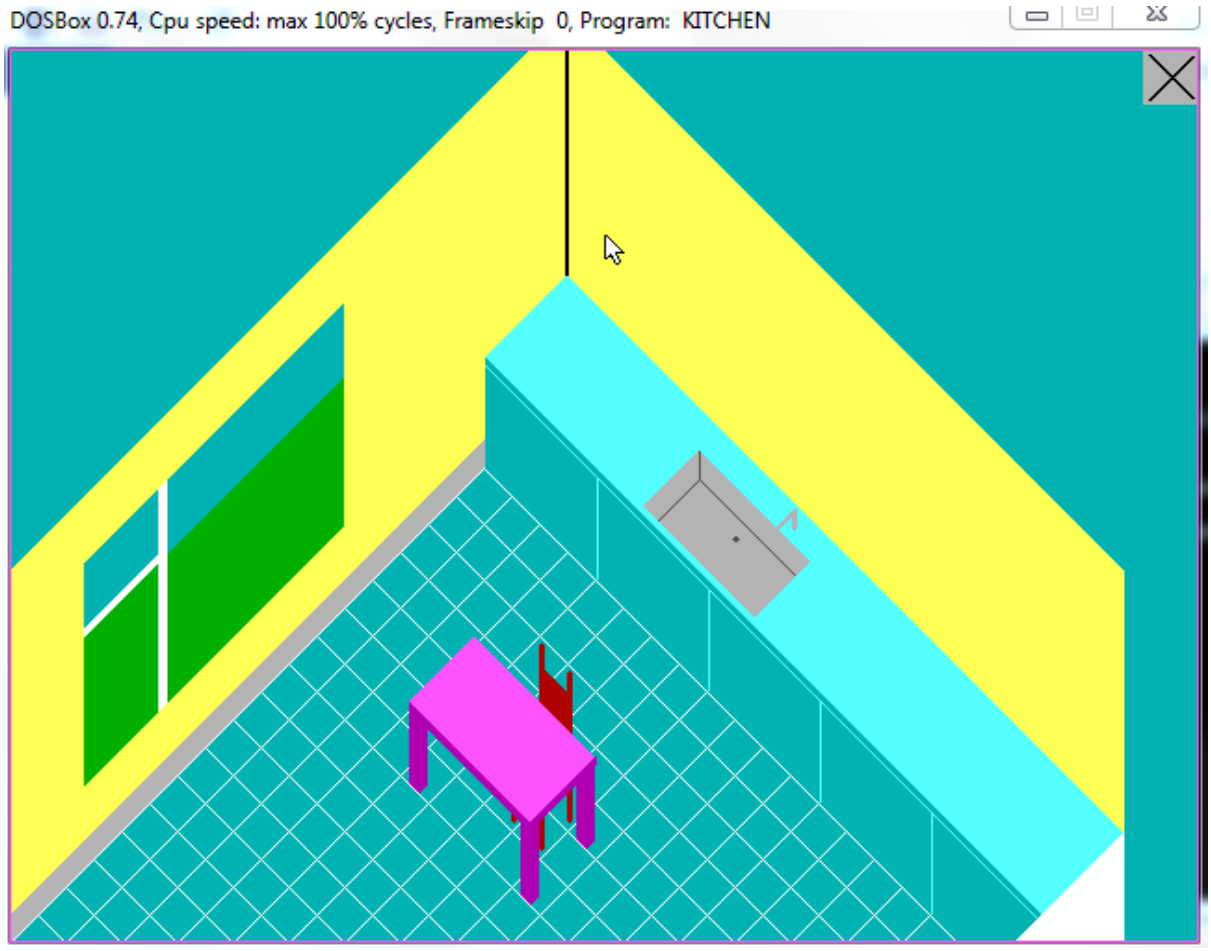
To create the chair moving, I had to draw the chair over itself in the background colour, then move back 45 degrees and redraw the tiles, over them draw the front of the counter, then draw the chair followed by the table. This redraws about half the scene to create the illusion of animation, but it doesn't look very good in motion.

Declaration of originality:

I, Conor Twomey, state that this project is 100% my own work, done to my own standard and approval. The project was to make some aspect of a kitchen or dining area scene with some animations and could be expanded with my own ideas. I used some reusable subroutines to make the progression and design of the project smoother.

Working screenshots:

Full Kitchen



Sink after click and before, there is an animation, but it goes too fast to capture

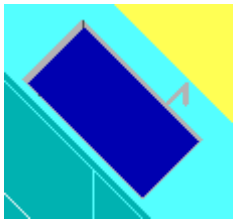
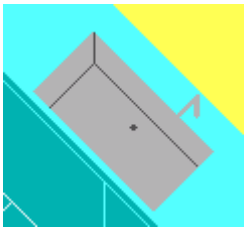
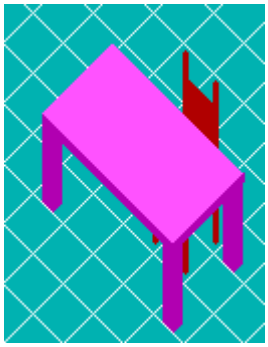
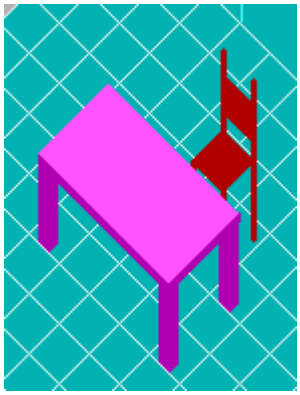



Table and chair before and after clicks, there is also an animation for this moving back slowly

Window, before click, during animation and after click



Complete code:

```
;
;       Author : Conor Twomey
;
;       File   : KITCHEN.ASM
;
;       This program is to draw a kitchen scene
;       using ASM. There are 3 onclick actions
;       All of the actions use animations

TITLE   KITCHEN.ASM

SSEG    SEGMENT PARA STACK 'STACK'        ;Stack segment
        DB      64 DUP('STACK   ')
SSEG    ENDS


DSEG    SEGMENT PARA PUBLIC 'DATA'        ;Code segment
DSEG    ENDS


CSEG    SEGMENT PARA PUBLIC 'CODE'
        ASSUME CS:CSEG, DS:DSEG, SS:SSEG

MAIN    PROC    FAR

        PUSH    DS                  ;These 5 lines are
        MOV     AX,0                 ;required for all .ASM
        PUSH    AX                  ;programs.
        MOV     AX,DSEG                 ;
```

```
    MOV     DS,AX                   ;


                                ;The program begins here.
    MOV     AH,00H                  ;Video: 640x480
    MOV     AL,18
    INT     10H

    CALL    DRAWSCENE
    CALL    DRAWXBUTTON
    MOV     AL,0DH
    CALL    DRAWFRAME
    CALL    MOUSEACTION


    MOV     AH,00H
    MOV     AL,2            ;Reset the screen to normal
    INT     10H


    RET                         ;The program ends here.

MAIN    ENDP

;-------------------------------------------------------------
;       DRAWXBUTTON
;       Description:    Draws X button to the screen
;       Input params:   None
;       Reg Effected:   None
;-------------------------------------------------------------
```

```
DRAWXBUTTON PROC
    PUSH    AX
    PUSH    CX
    PUSH    DX
    PUSH    SI
    PUSH    DI


    MOV     CX,610
    MOV     DX,0
    MOV     SI,30
    MOV     DI,30
    MOV     AL,07H
    CALL    FILLBOX                 ;Box above

    MOV     AL,00H                  ;X below
    ADD     CX,3
    ADD     DX,3
    MOV     SI,24
    CALL    DRAWDOWN45LINE
    INC     CX
    CALL    DRAWDOWN45LINE
    ADD     DX,24
    CALL    DRAWUP45LINE
    DEC     CX
    CALL    DRAWUP45LINE

    POP     DI
    POP     SI
    POP     DX
    POP     CX
```

```
    POP     AX

    RET
DRAWXBUTTON ENDP


;-------------------------------------------------------------
;       MOUSEACTION
;       Description:    Controls the mouse actions
;       Input params:   None
;       Reg Effected:   None
;-------------------------------------------------------------

MOUSEACTION PROC
    MOV     AX,01H                  ;Shows mouse cursor
    INT     33H

CHECKFORCLICK:

    MOV     AX,05H                  ;Checks for click
    MOV     BX,0H
    INT     33H
    CMP     BX,1
    JE      ACTION
    JMP     CHECKFORCLICK

ACTION:
    CALL    ONCLICK                 ;Performs click action
    CMP     CX,610                  ;Then checks for exit
    JB      CHECKFORCLICK
    CMP     DX,30
```

```
    JA      CHECKFORCLICK

    RET
MOUSEACTION ENDP

;-------------------------------------------------------------
;       ONCLICK
;       Description:    Handles mouse click
;       Input params:   None
;
;
;       Reg Effected:   CX - Column clicked
;                       DX - Row clicked
;-------------------------------------------------------------

ONCLICK PROC

OPTION1:


    CMP     CX,40               ;Checks for window area click
    JB      ENDCLICK
    CMP     CX,180
    JA      OPTION2
    CMP     DX,130
    JB      OPTION2
    CMP     DX,370
    JA      OPTION2

    MOV     AX,00H              ;Hides the cursor
    INT     33H                 ;Cancels overwritten colours
```

```
    MOV     AL,0FH
    CALL    DRAWBLINDS          ;Animates the blinds

    MOV     AX,01H              ;Reshows the mouse cursor
    INT     33H

    JMP     ENDCLICK

OPTION2:
    CMP     CX,340              ;Checks if sink area clicked
    JB      OPTION3
    CMP     CX,430
    JA      ENDCLICK
    CMP     DX,215
    JB      ENDCLICK
    CMP     DX,305
    JA      OPTION3

    MOV     AX,00H
    INT     33H

    CALL    TAPANIM             ;Animates filling sink
    MOV     AX,01H
    INT     33H

    JMP     ENDCLICK

OPTION3:
    CMP     CX,215              ;Check if table/chair area
```

```
    JB     ENDCLICK            ;clicked
    CMP    CX,317
    JA     ENDCLICK
    CMP    DX,300
    JB     ENDCLICK
    CMP    DX,460
    JA     ENDCLICK

    MOV    AX,00H
    INT    33H

    CALL   MOVECHAIR           ;Animates the moving chair

    MOV    AX,01H
    INT    33H

ENDCLICK:

    RET
ONCLICK ENDP

;-------------------------------------------------------------
;      MOVECHAIR
;      Description:    Animation for moving the chair
;      Input params:   None
;      Reg Effected:   None
;-------------------------------------------------------------

MOVECHAIR PROC
    PUSH   AX
    PUSH   BX
```

```
    PUSH    CX
    PUSH    DX

    MOV     CX,285
    MOV     DX,360
    MOV     BX,0
    MOV     AL,03H
    CALL    DRAWCHAIR                   ;Blanks out the chair

CHAIRMOVELOOP:
    CMP     BX,20                       ;Chair moving loop
    JE      FINISHMOVECHAIR
    INC     BX

    MOV     AL,03H
    ADD     CX,BX
    SUB     DX,BX
    CALL    DRAWCHAIR                   ;Blanks chair

    MOV     AL,0FH
    CALL    DRAWTILES
    CALL    DRAWCOUNTERFRONT            ;Redrawing tiles and counter

    INC     CX
    DEC     DX
    MOV     AL,04H
    CALL    DRAWCHAIR                   ;Redrawing chair

    MOV     AL,05H
    MOV     CX,280
    MOV     DX,420
```

```
    CALL    DRAWTABLE                   ;Redrawing table
                                        ;over the chair
    MOV     CX,285
    MOV     DX,360
    JMP     CHAIRMOVELOOP               ;Jumps back to start
                                        ;of the loop

FINISHMOVECHAIR:
    POP     DX
    POP     CX
    POP     BX
    POP     AX

    RET
MOVECHAIR ENDP

;---------------------------------------------------------------
;       TAPANIM
;       Description:    Animation for filling the sink
;       Input params:   None
;       Reg Effected:   None
;---------------------------------------------------------------

TAPANIM PROC
    PUSH    CX
    PUSH    DX
    PUSH    AX
    PUSH    BX
    PUSH    SI
    PUSH    DI
```

```
    MOV     CX,412                  ;Draw stream from tap
    MOV     DX,258
    MOV     AL,01H
    MOV     SI,20
    CALL    DRAWVLINE

    MOV     CX,350                  ;Preparing sink fill
    MOV     DX,253                  ;loop
    MOV     DI,22
    MOV     SI,51
    MOV     BX,6
FILLING:
    CMP     BX,0                    ;Loop for filling
    JE      FINISHSINK
    DEC     BX
    DEC     CX
    DEC     DX
    INC     SI
    INC     DI
    CALL    DRAWDIAMOND
    CALL    DELAY
    JMP     FILLING

FINISHSINK:
    MOV     CX,412
    MOV     DX,257
    MOV     AL,07H
    MOV     SI,3
    CALL    DRAWVLINE               ;Turn off the tap stream
```

```
    POP    DI
    POP    SI
    POP    BX
    POP    AX
    POP    DX
    POP    CX

    RET
TAPANIM ENDP

;-------------------------------------------------------------
;      DRAWBLINDS
;      Description:    DRAWBLINDS
;      Input params:   None
;      Reg Effected:   None
;-------------------------------------------------------------

DRAWBLINDS PROC
    PUSH   CX
    PUSH   DX
    PUSH   AX
    PUSH   DI
    PUSH   SI

    MOV    DI,117
    MOV    SI,140
    MOV    CX,40
    MOV    DX,275                  ;Preparing blind animation


AGAINBLIND:
```

```
    CMP     DI,0
    JE      FINISHBLINDS
    DEC     DI
    CALL    DRAWUP45LINE
    INC     DX
    CALL    DELAY
    JMP     AGAINBLIND                  ;Pulling blinds and looping

FINISHBLINDS:
    POP     SI
    POP     DI
    POP     AX
    POP     DX
    POP     CX

    RET
DRAWBLINDS ENDP



;-------------------------------------------------------------
;       DRAWSCENE
;       Description:    Draw scene to the screen
;       Input params:   None
;       Reg Effected:   None
;-------------------------------------------------------------

DRAWSCENE PROC
    PUSH    AX
    PUSH    BX
```

```
    PUSH    CX
    PUSH    DX
    PUSH    SI
    PUSH    DI

    MOV     CX,0
    MOV     DX,0
    MOV     SI,640
    MOV     AL,03H
    CALL    FILLBOX                 ;Blue background

    MOV     AL,0FH
    CALL    DRAWTILES               ;Draw tiles on floor
    MOV     AL,0EH
    CALL    DRAWWALLS               ;Draw walls of kitchen
    MOV     AL,0FH
    CALL    DRAWCOUNTER             ;Draw countertop of kitchen

    MOV     AL,04H
    MOV     CX,285
    MOV     DX,360
    CALL    DRAWCHAIR               ;Draw chair in kitchen

    MOV     AL,05H
    MOV     CX,280
    MOV     DX,420
    CALL    DRAWTABLE               ;Draw the table
    CALL    DRAWWINDOW              ;Draw the window

    POP     DI
    POP     SI
```

```
    POP     DX
    POP     CX
    POP     BX
    POP     AX

    RET
DRAWSCENE ENDP



;-------------------------------------------------------------
;       DRAWWINDOW
;       Description:    Draw window to the screen
;       Input params:   None
;       Reg Effected:   None
;-------------------------------------------------------------

DRAWWINDOW PROC
    PUSH    CX
    PUSH    DX
    PUSH    SI
    PUSH    DI
    PUSH    AX

    MOV     CX,40
    MOV     DX,395
    MOV     SI,140
    MOV     DI,120
    MOV     AL,03H
    CALL    DRAWVERTICALRHOMBUSREV          ;Sky outside window
```

```
    MOV     DI,80
    MOV     AL,02H
    CALL    DRAWVERTICALRHOMBUSREV          ;Grass outside window

    ADD     CX,40
    SUB     DX,40
    MOV     DI,120
    MOV     SI,5
    MOV     AL,0FH
    CALL    DRAWVERTICALRHOMBUSREV          ;Vertical window divisor

    SUB     CX,40
    SUB     DX,40
    MOV     DI,5
    MOV     SI,40
    CALL    DRAWVERTICALRHOMBUSREV          ;Horizontal window divisor

    POP     AX
    POP     DI
    POP     SI
    POP     DX
    POP     CX

    RET
DRAWWINDOW ENDP

;-------------------------------------------------------------
;       DRAWCOUNTERFRONT
;       Description:    Draw front of counter to the screen
;       Input params:   None
;       Reg Effected:   None
```

```
;-------------------------------------------------------------

DRAWCOUNTERFRONT PROC
    PUSH    AX
    PUSH    BX
    PUSH    CX
    PUSH    DX
    PUSH    SI
    PUSH    DI

    MOV     DI,60
    MOV     SI,300
    MOV     AL,03H
    MOV     CX,256
    MOV     DX,225
    CALL    DRAWVERTICALRHOMBUS          ;Draw front of counter

    SUB     DX,55
    MOV     AL,0BH
    MOV     SI,300
    CALL    DRAWDOWN45LINE               ;Top of cupboard doors
    MOV     BX,5
    MOV     SI,55


CUPBOARDLOOP:
    CMP     BX,0                         ;Loop for cupboard divisions
    JE      FINISHCUPBOARD

    ADD     CX,60
    ADD     DX,60
```

```
    CALL    DRAWVLINE

    DEC     BX
    JMP     CUPBOARDLOOP

FINISHCUPBOARD:

    POP     DI
    POP     SI
    POP     DX
    POP     CX
    POP     BX
    POP     AX

    RET
DRAWCOUNTERFRONT ENDP


;-------------------------------------------------------------
;       DRAWCOUNTER
;       Description:    Draw counter to the screen
;       Input params:   None
;       Reg Effected:   None
;-------------------------------------------------------------

DRAWCOUNTER PROC
    PUSH    DI
    PUSH    SI
    PUSH    CX
    PUSH    DX
    PUSH    AX
```

```
    MOV     CX,256
    MOV     DX,165
    MOV     SI,300
    MOV     DI,44
    MOV     AL,0BH
    CALL    DRAWDIAMOND                 ;Draw countertop

    ADD     CX,85
    ADD     DX,80
    MOV     SI,60
    MOV     DI,30
    CALL    DRAWSINK                    ;Draw sink on counter

    CALL    DRAWCOUNTERFRONT            ;Draw front of counter

    ADD     DX,110
    SUB     CX,56

    MOV     DX,580
    MOV     CX,500
    MOV     AL,0FH
    MOV     DI,61
    MOV     SI,100
    CALL    DRAWVERTICALRHOMBUSREV      ;Side end of counter


    POP     AX
    POP     DX
    POP     CX
```

```
    POP     SI
    POP     DI

    RET

DRAWCOUNTER ENDP

;---------------------------------------------------------------
;       DRAWSINK
;       Description:    Draw sink on the screen
;       Input params:   CX = Row
;                       DX = Column
;                       SI = Length
;                       DI = Width
;       Reg Effected:   None
;---------------------------------------------------------------
DRAWSINK PROC
    PUSH    AX
    PUSH    SI
    PUSH    CX
    PUSH    DX

    MOV     AL,07H
    CALL    DRAWDIAMOND                 ;Base shape of sink
    ADD     CX,DI
    SUB     DX,DI
    INC     DX
    MOV     SI,15
    MOV     AL,08H
    CALL    DRAWVLINE                   ;Back left corner
    ADD     DX,15
```

```
    MOV     SI,52
    CALL    DRAWDOWN45LINE              ;Bottom back of sink
    SUB     CX,22
    ADD     DX,22
    MOV     SI,22
    CALL    DRAWUP45LINE                ;Left bottom of sink
    ADD     CX,40
    ADD     DX,10
    MOV     SI,2
    MOV     DI,2
    CALL    DRAWDIAMOND                 ;Plughole of sink

                                ;Drawing tap
    ADD     CX,32
    SUB     DX,15
    MOV     SI,10
    MOV     AL,07H
    CALL    DRAWVLINE                   ;Back of tap
    INC     CX
    INC     DX
    CALL    DRAWVLINE
    INC     CX
    DEC     DX
    CALL    DRAWVLINE

    INC     SI
    SUB     CX,10
    ADD     DX,10
    CALL    DRAWUP45LINE                ;Top of tap, horizontal
    DEC     DX
    CALL    DRAWUP45LINE
```

```
    DEC    CX
    CALL   DRAWUP45LINE


    POP    DX
    POP    CX
    POP    SI
    POP    AX

    RET
DRAWSINK ENDP


;-------------------------------------------------------------
;      DRAWWALLS
;      Description:    Draw walls to the screen
;      Input params:   AX - Colour of walls
;      Reg Effected:   None
;-------------------------------------------------------------

DRAWWALLS PROC
    PUSH   AX
    PUSH   CX
    PUSH   DX
    PUSH   DI
    PUSH   SI

    MOV    CX,300
    MOV    DX,180
    MOV    SI,300
    MOV    DI,200
```

```
    CALL    DRAWVERTICALRHOMBUS                 ;Draw right wall
    MOV     CX,0
    MOV     DX,479
    CALL    DRAWVERTICALRHOMBUSREV              ;Draw left wall
    MOV     DI,15
    MOV     AL,07H
    CALL    DRAWVERTICALRHOMBUSREV              ;Skirting board
    ADD     CX,299
    SUB     DX,480
    MOV     AL,00H
    MOV     SI,200
    CALL    DRAWVLINE                           ;Vertical line
    ADD     CX,1                                ;Helps to define
    CALL    DRAWVLINE                           ;the corner


    POP     SI
    POP     DI
    POP     DX
    POP     CX
    POP     AX

    RET

DRAWWALLS ENDP


;-------------------------------------------------------------
;       DRAWTILES
;       Description:    Draw tiles to the screen
;       Input params:   None
```

```
;       Reg Effected:   None
;-------------------------------------------------------------

DRAWTILES PROC

        PUSH    CX
        PUSH    DX
        PUSH    AX
        PUSH    DI
        PUSH    SI

        MOV     CX,0
        MOV     DX,480
        MOV     DI,18
        MOV     SI,30
TILELOOP:
        CMP     DI,0
        JE      TILESP2                 ;When complete, move to
        DEC     DI                      ;up 45 lines

        CALL    DRAWDOWN45LINE          ;Draw down 45 lines
        ADD     CX,15
        SUB     DX,15
        ADD     SI,15
        JMP     TILELOOP                ;Back for another row

TILESP2:MOV     DI,18
        MOV     SI,270
        MOV     CX,0
        MOV     DX,480                  ;reset values for part 2
TILELOOP2:
```

```
    CMP     DI,0
    JE      ENDTILES                    ;finish function
    DEC     DI

    CALL    DRAWUP45LINE
    ADD     CX,30
    SUB     SI,15
    JMP     TILELOOP2


ENDTILES:
    POP     SI
    POP     DI
    POP     AX
    POP     DX
    POP     CX

    RET

DRAWTILES ENDP



;-------------------------------------------------------------
;       DRAWCHAIR
;       Description:    Draw chair to the screen
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;       Reg Effected:   None
;-------------------------------------------------------------
```

```
DRAWCHAIR PROC

    PUSH    CX
    PUSH    DX
    PUSH    AX
    PUSH    SI

    MOV     SI,40                   ;Length of half chair leg

    CALL    DRAWFULLCHAIRLEG        ;Back right chair leg
                                    ;From sitting on it
    ADD     CX,15
    ADD     DX,15
    CALL    DRAWFULLCHAIRLEG        ;Back left leg

    ADD     DX,15
    SUB     CX,15

    CALL    DRAWHALFCHAIRLEG        ;Front right leg

    SUB     CX,15
    SUB     DX,15
    CALL    DRAWHALFCHAIRLEG        ;Front left leg

    MOV     DI,16
    MOV     SI,16
    CALL    DRAWDIAMOND             ;Seat of chair

    ADD     CX,15
    SUB     DX,25
```

```
    MOV     DI,20
    CALL    DRAWVERTICALRHOMBUS          ;Back of chair

    POP     SI
    POP     AX
    POP     DX
    POP     CX

    RET

DRAWCHAIR ENDP



;---------------------------------------------------------------
;       DRAWFULLCHAIRLEG
;       Description:    Draw leg of chair to the screen
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;                       SI = Height of leg
;       Reg Effected:   None
;---------------------------------------------------------------

DRAWFULLCHAIRLEG PROC

    PUSH    CX
    PUSH    DX
    PUSH    SI

    CALL    DRAWHALFCHAIRLEG
```

```
    SUB     DX,SI
    CALL    DRAWHALFCHAIRLEG

    POP     SI
    POP     DX
    POP     CX

    RET

DRAWFULLCHAIRLEG ENDP



;-------------------------------------------------------------
;       DRAWHALFCHAIRLEG
;       Description:    Draw leg of chair to the screen
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;                       SI = Height of leg
;       Reg Effected:   None
;-------------------------------------------------------------

DRAWHALFCHAIRLEG PROC

    PUSH    CX
    PUSH    DX
    PUSH    SI

    CALL    DRAWVLINE
```

```
    ADD    CX,2
    CALL   DRAWVLINE
    SUB    CX,1
    SUB    DX,1
    ADD    SI,2
    CALL   DRAWVLINE

    POP    SI
    POP    DX
    POP    CX

    RET

DRAWHALFCHAIRLEG ENDP



;---------------------------------------------------------------
;      DRAWTABLE
;      Description:    Draw top of table to the screen
;      Input params:   DX = Row
;                      CX = Column
;                      AL = Colour
;      Reg Effected:   None
;---------------------------------------------------------------

DRAWTABLE PROC

    PUSH   CX
    PUSH   DX
```

```
    PUSH    SI

    MOV     SI,40
    CALL    DRAWTABLELEG

    SUB     DX,60
    SUB     CX,60
    CALL    DRAWTABLELEG

    ADD     DX,30
    ADD     CX,90
    CALL    DRAWTABLELEG

    SUB     CX,95
    SUB     DX,35
    CALL    DRAWTABLETOP

    POP     SI
    POP     DX
    POP     CX

    RET

DRAWTABLE ENDP



;-------------------------------------------------------------
;       DRAWTABLETOP
;       Description:    Draw top of table to the screen
```

```
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;       Reg Effected:   None
;-------------------------------------------------------------

DRAWTABLETOP PROC

        PUSH    CX
        PUSH    DX
        PUSH    SI
        PUSH    DI
        PUSH    AX

        MOV     SI,66
        MOV     DI,5
TTOP1:  CMP     DI,0
        JE      TTOP2S
        CALL    DRAWDOWN45LINE              ;Draw left side of top
        DEC     DX
        DEC     DI
        JMP     TTOP1

TTOP2S: MOV     DI,35
        ADD     AL,08H
TTOP2:  CMP     DI,0
        JE      TTOP3S
        CALL    DRAWDOWN45LINE              ;Draw top of table
        DEC     DI
        INC     CX
        CALL    DRAWDOWN45LINE
```

```
    DEC   DX
    JMP   TTOP2


TTOP3S: MOV   DI,5
    MOV   SI,36
    ADD   CX,30
    ADD   DX,100
    SUB   AL,08H

TTOP3:  CMP   DI,0
    JE    TTOPFINISH              ;Draw front/right
    CALL  DRAWUP45LINE
    INC   DX
    DEC   DI
    JMP   TTOP3

TTOPFINISH:

    POP   AX
    POP   DI
    POP   SI
    POP   DX
    POP   CX

    RET

DRAWTABLETOP ENDP
```

```
;-------------------------------------------------------------
;       DRAWTABLELEG
;       Description:    Draw leg of table to the screen
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;       Reg Effected:   None
;-------------------------------------------------------------

DRAWTABLELEG PROC

        PUSH    CX
        PUSH    DX
        PUSH    SI
        PUSH    DI
        PUSH    BX

        MOV     DI,SI

        MOV     SI,5

        MOV     SI,DI
        ADD     CX,5
        SUB     DX,5
        MOV     BX,5

TLEG1:
        CMP     BX,0
        JE      TLEG2
        DEC     BX
```

```
    DEC    CX
    INC    DX
    CALL   DRAWVLINE                ;Draw right side of leg
    JMP    TLEG1

TLEG2:
    SUB    CX,6
    SUB    DX,6
    MOV    BX,5

TLEG_LOOP:
    CMP    BX,0
    JE     TLEGFINISH
    DEC    BX

    INC    CX
    INC    DX
    CALL   DRAWVLINE                ;Draw left side of leg
    JMP    TLEG_LOOP


TLEGFINISH:
    POP    BX
    POP    DI
    POP    SI
    POP    DX
    POP    CX

    RET

DRAWTABLELEG ENDP
```

```
;-------------------------------------------------------------
;       WRITEPIXEL
;       Description:    Write pixel to the screen
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;       Reg Effected:   None
;-------------------------------------------------------------

WRITEPIXEL      PROC
        PUSH    AX
        PUSH    BX

        MOV     AH,0CH
        MOV     BH,00H
        INT     10H

        POP     BX
        POP     AX
        RET

WRITEPIXEL      ENDP

;-------------------------------------------------------------
;       DRAWHLINE
;       Description:    Draw horizontal line of pixels
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;                       SI = Number of pixels in line
```

```
;       Reg Effected:   None
;-------------------------------------------------------------


DRAWHLINE       PROC
        PUSH    CX
        PUSH    SI

AGAIN_DHL:
        CMP     SI,0
        JE      FINISH_DHL
        CALL    WRITEPIXEL
        DEC     SI
        INC     CX
        JMP     AGAIN_DHL


FINISH_DHL:

        POP     SI
        POP     CX

        RET

DRAWHLINE       ENDP
;-------------------------------------------------------------


;-------------------------------------------------------------
;       DRAWDIAMOND
;       Description:    Draw diamond to the screen
```

```
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;                       SI = Length
;                       DI = Width
;       Reg Effected:   None
;-------------------------------------------------------------

DRAWDIAMOND PROC

        PUSH    CX
        PUSH    DX
        PUSH    SI
        PUSH    DI

DIAM_1: CMP     DI,0
        JE      DIAMFINISH
        DEC     DI

        CALL    DRAWDOWN45LINE
        INC     CX
        CALL    DRAWDOWN45LINE
        DEC     DX
        JMP     DIAM_1

DIAMFINISH:
        POP     DI
        POP     SI
        POP     DX
        POP     CX
```

```
    RET

DRAWDIAMOND ENDP


;-------------------------------------------------------------
;   DRAWVERTICALRHOMBUS
;   Description:    Draw vertical rhombus to the screen
;   Input params:   DX = Row
;                   CX = Column
;                   AL = Colour
;                   SI = Width
;                   DI = Height
;   Reg Effected:   None
;-------------------------------------------------------------

DRAWVERTICALRHOMBUS PROC

    PUSH    CX
    PUSH    DX
    PUSH    SI
    PUSH    DI

VRHOM:  CMP     DI,0
    JE      VRHOMFINISH
    DEC     DI

    CALL    DRAWDOWN45LINE
    DEC     DX
    JMP     VRHOM
```

```
VRHOMFINISH:
    POP    DI
    POP    SI
    POP    DX
    POP    CX

    RET

DRAWVERTICALRHOMBUS ENDP

;-------------------------------------------------------------
;    DRAWVERTICALRHOMBUSREV
;    Description:    Draw reverse vertical rhombus to the screen
;    Input params:   DX = Row
;                    CX = Column
;                    AL = Colour
;                    SI = Width
;                    DI = Height
;    Reg Effected:   None
;-------------------------------------------------------------

DRAWVERTICALRHOMBUSREV PROC

    PUSH   CX
    PUSH   DX
    PUSH   SI
    PUSH   DI

VRHOMR: CMP    DI,0
    JE     VRHOMRFINISH
    DEC    DI
```

```
    CALL    DRAWUP45LINE
    DEC     DX
    JMP     VRHOMR

VRHOMRFINISH:
    POP     DI
    POP     SI
    POP     DX
    POP     CX

    RET

DRAWVERTICALRHOMBUSREV ENDP


;---------------------------------------------------------------
;       DRAWVLINE
;       Description:    Draw vertical line of pixels
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;                       SI = Number of pixels in line
;       Reg Effected:   None
;---------------------------------------------------------------

DRAWVLINE       PROC
    PUSH    DX
    PUSH    SI
```

```
AGAIN_DVL:
    CMP    SI,0
    JE     FINISH_DVL
    CALL   WRITEPIXEL
    DEC    SI
    INC    DX
    JMP    AGAIN_DVL


FINISH_DVL:

    POP    SI
    POP    DX

    RET

DRAWVLINE      ENDP

;---------------------------------------------------------------
;      FILLBOX
;      Description:    Draw full colur box
;      Input params:   DX = Row
;                      CX = Column
;                      AL = Colour
;                      SI = Number of pixels in line
;                      DI = Height of box
;      Reg Effected:   None
;---------------------------------------------------------------
FILLBOX PROC

    PUSH   DI
```

```
    PUSH    DX

FBOX_LOOP:
    CMP     DI,0
    JE      FBOX_FINISH
    DEC     DI
    CALL    DRAWHLINE
    INC     DX
    JMP     FBOX_LOOP


FBOX_FINISH:
    POP     DX
    POP     DI

    RET
FILLBOX ENDP

;---------------------------------------------------------------
;       DRAWUP45LINE
;       Description:    Draw diagonal line of pixels
;                       Bottom left to top right
;       Input params:   DX = Row
;                       CX = Column
;                       AL = Colour
;                       SI = Number of pixels in line
;       Reg Effected:   None
;---------------------------------------------------------------


DRAWUP45LINE       PROC
```

```
    PUSH    CX
    PUSH    DX
    PUSH    SI

AGAIN_DUDL:
    CMP     SI,0
    JE      FINISH_DUDL
    CALL    WRITEPIXEL
    DEC     SI
    INC     CX
    DEC     DX
    JMP     AGAIN_DUDL


FINISH_DUDL:

    POP     SI
    POP     DX
    POP     CX

    RET

DRAWUP45LINE      ENDP
;---------------------------------------------------------
;---------------------------------------------------------
;       DRAWDOWN45LINE
;       Description:    Draw diagonal line of pixels
;                       Top left to bottom right
;       Input params:   DX = Row
```

```
;                   CX = Column
;                   AL = Colour
;                   SI = Number of pixels in line
;       Reg Effected:   None
;-----------------------------------------------------------------


DRAWDOWN45LINE      PROC
        PUSH    CX
        PUSH    DX
        PUSH    SI

AGAIN_DDDL:
        CMP     SI,0
        JE      FINISH_DDDL
        CALL    WRITEPIXEL
        DEC     SI
        INC     CX
        INC     DX
        JMP     AGAIN_DDDL


FINISH_DDDL:

        POP     SI
        POP     DX
        POP     CX

        RET

DRAWDOWN45LINE      ENDP
```

```
;-------------------------------------------------------------



;-------------------------------------------------------------
;       DELAY
;       Description:    Delay
;       Input params:
;       Reg Effected:   None
;-------------------------------------------------------------

DELAY   PROC

        PUSH    CX
        PUSH    DX

        MOV     CX,30H

OUTERLOOP:
        DEC     CX
        CMP     CX,0
        JE      FINISH_DELAY

        MOV     DX,0FFFFH
INNERLOOP:
        DEC     DX
        CMP     DX,0
        JE      OUTERLOOP
        JMP     INNERLOOP

FINISH_DELAY:
```

```
    POP     DX
    POP     CX
    RET

DELAY   ENDP

;---------------------------------------------------------
;       DRAWFRAME
;       Description:    Draw frame to the screen
;       Input params:   None
;       Reg Effected:   None
;---------------------------------------------------------
DRAWFRAME PROC
    PUSH    CX
    PUSH    DX
    PUSH    SI

    MOV     SI,640
    MOV     CX,0
    MOV     DX,0
    CALL    DRAWHLINE
    MOV     DX,479
    CALL    DRAWHLINE

    MOV     DX,0
    MOV     SI,480
    CALL    DRAWVLINE
    MOV     CX,639
    CALL    DRAWVLINE
```

```
    POP    SI
    POP    DX
    POP    CX
    RET
DRAWFRAME ENDP

CSEG   ENDS
    END    MAIN
```