Draw a House with EMU8086

Assembly Language Coding Project

Matthew Newcomer

EE 3612

Professor John Helferty

October 18, 2010

Table of Contents

Matthew Newcomer
EE 3612 10/18/10
Housing Graphic Project

We were assigned a problem to use the EMU8086 emulator's graphical display, simulating assembly language on an 8086 microprocessor. The class was supposed to write code that would make a pixel drawing of a house, with a door and two windows. In addition to the house, we needed to draw a path leading to the door of the house and a tree adjacent to the house. A representation of the sun was to be included if possible.

I began this assignment by drawing a pixel field on a piece of paper so I could easily establish specific components of the graphic and the pixel coordinates of each element. Working with slides from class, I setup the emulator for the graphics display. Using examples from the slides, I experimented until I was proficient at drawing single linear lines. The house started with four lines: top, bottom, left and right. Using specifically defined starting and ending coordinates, I drew straight lines for the door and windows of the house. The pitched roof was drawn by incrementing (inc) and decreasing (dec) both column and row coordinate positions at the same time. I made the lines comprising the house light grey (07h) to easily standout on the default black emulator screen. The path was made light red (0ch), the tree trunk was drawn brown (06h); both the path and tree trunk were drawn with a combination of slanted and vertical lines. The same coding methods were used for the house, path and tree trunk.

Drawing the green "leaves" of the tree was more complicated than simply drawing straight lines. The green tree could have been drawn with specific lines declared in the same way as the previous elements, but that process would have been time consuming and not ideal. I used implied loop statements and assorted jump commands to draw the green field of the tree in a triangular shape. The code starts at the bottom of the green tree (the longest green line) and draws the green line. The specific loop used to draw each line is similar to code used to draw a single straight line. The column position was set with the *cx* register. The bx register was used to store a length value of the longest green line. As the green tree loop repeats the row position is moved up while *cx* is shifted so the next line is drawn to the right. At the same the *bx* register is reduced to make a shorter line. The loop repeats automatically drawing the next green line shorter than the last, creating a pointed "pine tree". The sun was drawn with a similar method as the green part of the tree. The sun has three different nested loop statements, and assorted jump commands. The top of the sun is drawn from a starting point of the top left pixel of the sun. The first section of the sun is drawn with a yellow line that gets longer with each pass. The middle of the sun creates a rectangular yellow block. The bottom of the sun has code which decreases the size of the yellow line with each pass in a very similar process to the way the green tree was drawn. I used the *ds* register to store the value of *cx* to retain a count value of *cx* as each pass through the loop was executed. Storing the value of *cx* in the *ds* register simplified my process greatly, from my initial technique used for the green tree.

Assembly Language Code

```
org 100h

; clear the screen

mov ax,0600h    ;scroll the screen
mov bh,07       ;normal attribute
mov cx,0000     ;from row=00,column=00
mov dx,184fh    ;to row=18h, column=4fh
int 10h         ;invoke the interrupt to clear screen

mov ah,00       ;set mode
mov al,13h      ;mode=13(CGA High resolution)
int 10h         ;invoke the interrupt to change
mode


    ; Start drawing house

; top horizontal line (house)

mov cx,130      ;start line at column=130 and
mov dx,75       ;row=75
hseT: mov ah,0ch ;ah=0ch to draw a line
mov al,07h      ;pixels=light grey
int 10h         ;invoke the interrupt to draw the
line
inc cx          ;increment the horizontal position
cmp cx,216      ;draw line until column=216
jnz hseT

; bottom horizontal line (house)

mov cx,130
mov dx,125
hseB: mov ah,0ch
mov al,07h
int 10h
inc cx
cmp cx,216
jnz hseB

; left vertical line (house)

mov cx,130
mov dx,75
hseL: mov ah,0ch
mov al,07h
int 10h
inc dx
cmp dx,125
jnz hseL

; right vertical line (house)

mov cx,216
mov dx,75
hseR: mov ah,0ch
mov al,07h
int 10h
inc dx
cmp dx,126
jnz hseR

; left roof line (house)

mov cx,130
mov dx,75
hseLR: mov ah,0ch
mov al,07h
int 10h
inc cx
dec dx
cmp cx,173
cmp dx,32
jnz hseLR

; right roof line (house)

mov cx,173
mov dx,32
hseRR: mov ah,0ch
mov al,07h
int 10h
inc cx
inc dx
cmp cx,216
cmp dx,75
jnz hseRR

    ;!!House Outline Finished!!


           ; Draw the Door

; left door line (house)

mov cx,164
mov dx,125
hseLD: mov ah,0ch
mov al,07h
int 10h
dec dx
cmp dx,100
jnz hseLD
```

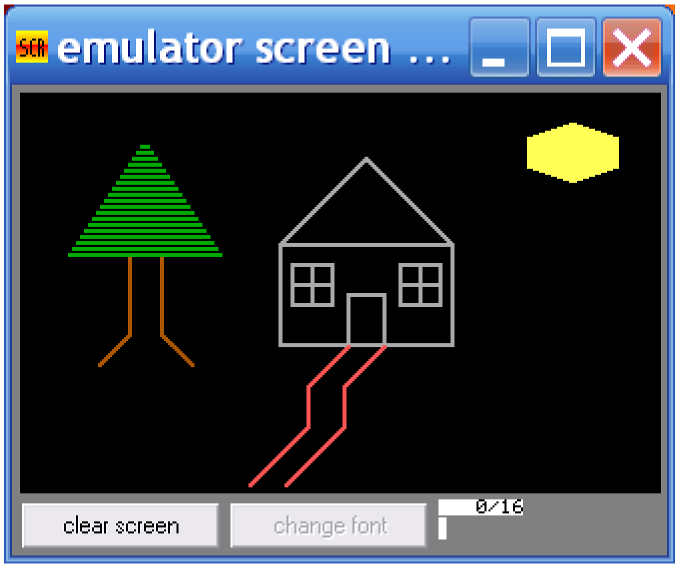
```
; right door line (house)

mov cx,182
mov dx,125
hseRD: mov ah,0ch
mov al,07h
int 10h
dec dx
cmp dx,100
jnz hseRD

; top door line (house)

mov cx,164
mov dx,100
hseTD: mov ah,0ch
mov al,07h
int 10h
inc cx
cmp cx,183
jnz hseTD

    ;!!Door Finished!!

              ; Draw two Windows

; left window vert line1 (house)

mov cx,136
mov dx,85
hseLWV1: mov ah,0ch
mov al,07h
int 10h
inc dx
cmp dx,105
jnz hseLWV1

; left window vert line2 (house)

mov cx,146
mov dx,85
hseLWV2: mov ah,0ch
mov al,07h
int 10h
inc dx
cmp dx,105
jnz hseLWV2

;left window vert line3 (house)

mov cx,156
mov dx,85
hseLWV3: mov ah,0ch
mov al,07h
int 10h
inc dx
cmp dx,105
jnz hseLWV3

; right window vert line1 (house)

mov cx,190
mov dx,85
hseRWV1: mov ah,0ch
mov al,07h
int 10h
inc dx
cmp dx,105
jnz hseRWV1

; right window vert line2 (house)

mov cx,200
mov dx,85
hseRWV2: mov ah,0ch
mov al,07h
int 10h
inc dx
cmp dx,105
jnz hseRWV2

; left window vert line3 (house)

mov cx,210
mov dx,85
hseRWV3: mov ah,0ch
mov al,07h
int 10h
inc dx
cmp dx,105
jnz hseRWV3

; window horz line1 (house)

mov cx,136
mov dx,85
hseWH1: mov ah,0ch
mov al,07h
int 10h
inc cx
cmp cx,156
jnz hseWH1

mov cx,190  ;this line continues for the second
window at column=190
mov dx,85
hseWH1b: mov ah,0ch
mov al,07h
int 10h
inc cx
cmp cx,210
jnz hseWH1b
```

```
; window horz line2 (house)

mov cx,136
mov dx,95
hseWH2: mov ah,0ch
mov al,07h
int 10h
inc cx
cmp cx,156
jnz hseWH2

mov cx,190
mov dx,95
hseWH2b: mov ah,0ch
mov al,07h
int 10h
inc cx
cmp cx,210
jnz hseWH2b

; window horz line3 (house)

mov cx,136
mov dx,105
hseWH3: mov ah,0ch
mov al,07h
int 10h
inc cx
cmp cx,157
jnz hseWH3

mov cx,190
mov dx,105
hseWH3b: mov ah,0ch
mov al,07h
int 10h
inc cx
cmp cx,211
jnz hseWH3b

    ;Windows Finished

    ;!!House Finished!!

    ; Draw the path

; left path top  (path)

mov cx,164
mov dx,126
pthLT: mov ah,0ch
mov al,0ch       ;pixels=light red
int 10h
dec cx
inc dx
cmp cx,144
cmp dx,146
jnz pthLT

; right path top (path)

mov cx,182
mov dx,126
pthRT: mov ah,0ch
mov al,0ch
int 10h
dec cx
inc dx
cmp cx,162
cmp dx,146
jnz pthRT

; left path middle  (path)

mov cx,144
mov dx,146
pthLM: mov ah,0ch
mov al,0ch
int 10h
inc dx
cmp dx,166
jnz pthLM

; right path middle (path)

mov cx,162
mov dx,146
pthRM: mov ah,0ch
mov al,0ch
int 10h
inc dx
cmp dx,166
jnz pthRM

; left path bottom  (path)

mov cx,144
mov dx,166
pthLB: mov ah,0ch
mov al,0ch
int 10h
dec cx
inc dx
cmp cx,114
cmp dx,196
jnz pthLB
```

```
; right path bottom (path)

mov cx,162
mov dx,166
pthRB: mov ah,0ch
mov al,0ch
int 10h
dec cx
inc dx
cmp cx,132
cmp dx,196
jnz pthRB

    ;!!Path Finished!!

    ; Draw the tree

; left trunk base (tree)

mov cx,40
mov dx,135
treLTB: mov ah,0ch
mov al,06h        ;pixels=brown
int 10h
inc cx
dec dx
cmp cx,55
cmp dx,120
jnz treLTB

; right trunk base (tree)

mov cx,86
mov dx,135
treRTB: mov ah,0ch
mov al,06h
int 10h
dec cx
dec dx
cmp cx,71
cmp dx,120
jnz treRTB

; left trunk vert (tree)

mov cx,55
mov dx,120
treLTV: mov ah,0ch
mov al,06h
int 10h
dec dx
cmp dx,80
jnz treLTV

; right trunk vert (tree)

mov cx,71
mov dx,120
treRTV: mov ah,0ch
mov al,06h
int 10h
dec dx
cmp dx,80
jnz treRTV

; I like pine trees

mov dx,80
mov bx,101

DrwTree:       ;draws a green pine tree with a 3
nested jump commands

mov cx,126
sub cx,bx      ;keeps green tree symmetrical

GrTree: mov ah,0ch
mov al,02h       ;pixels=green
int 10h
inc cx
cmp cx,bx
jnz GrTree

cmp bx,65        ;when bx reaches a
predetermined value (size of green tree)
jle break        ;loop exits

sub dx,3         ;moves green line up
sub bx,2         ;makes tree narrow on each pass

jmp DrwTree

break:

    ;!!Tree Finished!!

    ; Draw the sun

mov dx,15        ;establish initial position of sun
(top left pixel)
mov bx,278
mov cx,278
mov ds,cx
```

```
;draw the top half of the sun
DrwSunT:

mov cx,ds        ;cx gets cleared when the loop
repeats so value needs to be stored
sub cx,3         ;start the yellow line further to the
left each pass
mov ds,cx        ;stores cx value for next pass

DrwSunTp: mov ah,0ch
mov al,0eh       ;pixels=yellow
int 10h
inc cx
cmp cx,bx
jnz DrwSunTp

cmp dx,22        ;when the sun has drawn 7 rows
breaks to the middle section
je break2

inc dx           ;move the line to be colored yellow
down one row
add bx,3         ;makes the sun wider each pass

jmp DrwSunT      ;repeats the loop with new dx,
bx and cx values

break2:

;draw sun middle
DrwSunM:

inc dx           ;draws a rectangular middle section
mov cx,ds        ;remembers how long to make the
yellow line on each pass

DrwSunMp: mov ah,0ch
mov al,0eh
int 10h
inc cx
cmp cx,bx
jnz DrwSunMp

cmp dx,36        ;jumps to the bottom section
when 13 rect rows have been drawn
je break3

jmp DrwSunM

break3:

;draw bottom of sun
DrwSunB:

inc dx           ;similar to top part of sun
mov cx,ds
add cx,3         ;each pass line starts further to the
right
mov ds,cx
sub bx,3         ;each pass line gets shorter


DrwSunBp: mov ah,0ch
mov al,0eh
int 10h
inc cx
cmp cx,bx
jnz DrwSunBp

cmp dx,43        ;after 7 more rows each narrower
than prev (upside triangle)
je breakfinal    ;graphic program jumps for the
last time

jmp DrwSunB:

breakfinal:      ;returns control to the op. system

   ;!!Sun Finished!!

   ;!!!! ALL FINISHED !!!!


ret
```

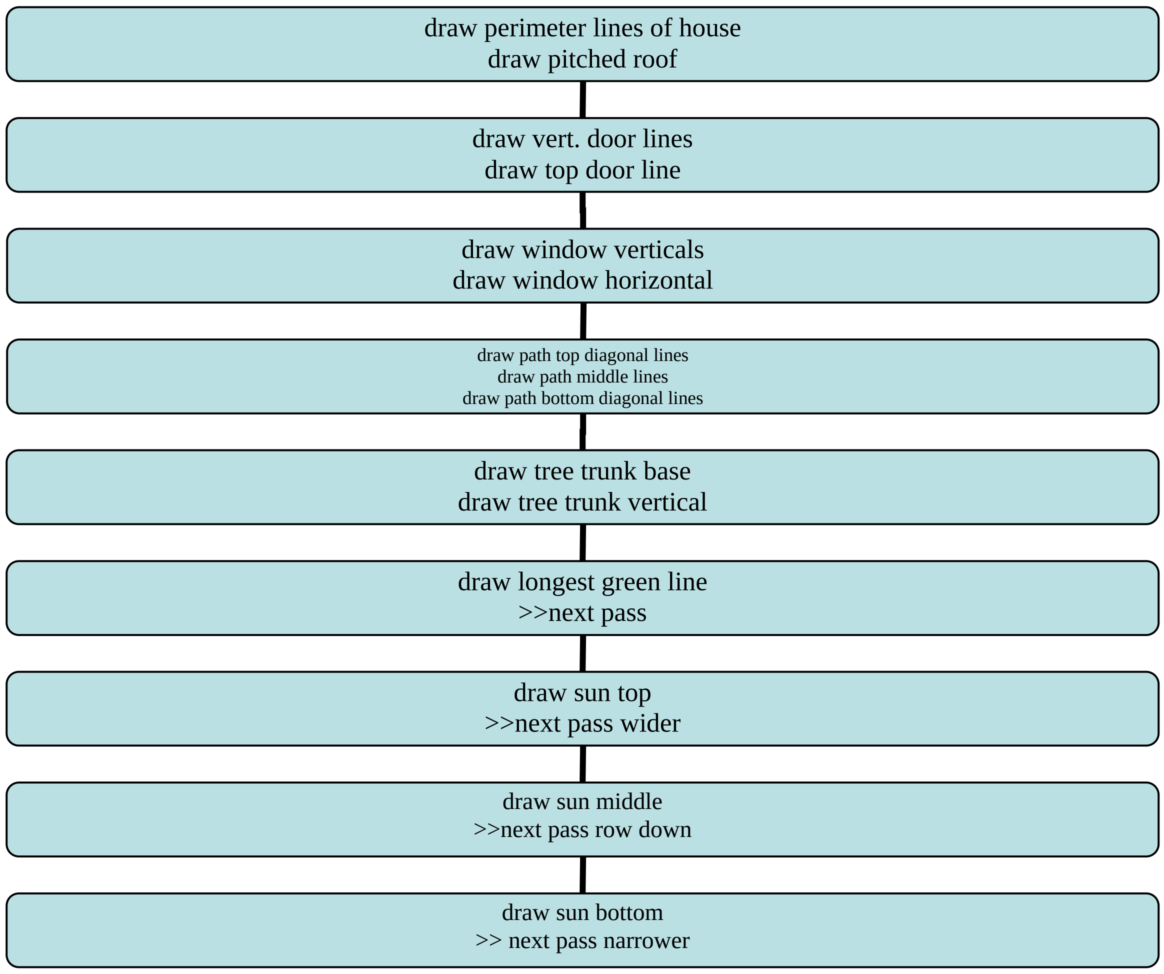
Screenshot of House Graphic



Color Key
house lines = light gray
path = light red
tree trunk = brown
tree leaves = green
sun = yellow

This program took 8 minutes 40 seconds to run and has over 500 lines of code.

Organizational Flowchart


draw perimeter lines of house
draw pitched roof
draw vert. door lines
draw top door line
draw window verticals
draw window horizontal
draw path top diagonal lines
draw path middle lines
draw path bottom diagonal lines
draw tree trunk base
draw tree trunk vertical
draw longest green line
>>next pass
draw sun top
>>next pass wider
draw sun middle
>>next pass row down
draw sun bottom
>> next pass narrower