CEDAR VALLEY 99er UG OCTOBER 1993

PRESIDENT:	Jack Johns	(319) 366-4541
VICE PRES:	Bob Wahlstrom	(319) 393-6042
TREASURER:	Bruce Winter	(319) 393-0610
SECRETARY:	Wayne Betts	(319) 377-2493
NL EDITOR:	Jeff Craft	(319) 393-6213
LIBRARIAN:	Bob Heiderstadt	(319) 927-4215

CEDAR RAPIDS/MARION

Supporting the TI-99/4A and 9640 in Eastern Iowa for over 10 years!

NEXT MEETING: 6:30 PM OCT 12, 1993

WEST MUSIC, COLLINS ROAD PLAZA

CONTENTS

Start the presses! After a month's delay, the Cedar Valley 99er newsletter is up and running again. Yes, that's right, Gary finally reeled one in! With my (HI-HO) black and silver console, 2 DSSD Drives, a half loaded P-box, and my basic but trusty TI impact printer, I have decided to give it a shot. It's going to be a real learning experience for me as I have never used the formatter before. I have owned a TI console since about 1983, but I didn't purchase the other items until about a year and a half ago. My main uses so far have been title screens for videos using TI-ARTIST and making labels.

On behalf of the Cedar Valley 99ers and probably many others I would like to thank Gary Bishop for all his efforts in the writing and producing of this newsletter. Gary promises to keep me well supplied with articles but "HEY YOU"!!, we need your input too. Let's keep the presses rolling. THANKS!

JEFF CRAFT

The Prez's Blurb

Trivia..How close does a magnet have to be to a computer disk to damage it?

Newsletters..Because Jim Green has a shelf with three feet of our old exchange newsletters, we voted at the last meeting that all newsletters more that one year old will be thrown in the recycling bin. If you have a need for any that are older see Jim before one month after you receive this newsletter. Any newer than one year will go to me.

Game Modules..I am looking for the games Arcturus and Buck Rogers in module form. I will pay $10 each.

RS232's..In a previous newsletter I described a way to let my modem work with my Myarc RS232. This month I found that it also works on a Corcomp RS232. For this reason I am repeating it here. The above mentioned RS232's may not handle the voltage levels coming from some modems, even though the RS232 spec states they should. To correct for this defect a 4.7K ohm resistor must be placed in the wire comming from the MODEM END pin number 3. This is the TX line. The resistor will limit the level to the RS232.

TI Wake..Despite underwhelming interest at the last meeting I do invite anyone who desires to wear black to the next meeting. This is to commemorate the 10 year anniversary of TI's orphaning of the TI-994/A.

Genealogy..Because a high percentage of those present at the last meeting acknowledged an interest in genealogy, I invite anyone with genealogy info, programs, talks, etc. to share them with us at the next meeting. If time does not permit everyone at the next meeting, we will have it at the one after. Also if you need time to prepare, the following meeting will have at least some time available for you.

Misc..I recommend the article in the last Popular Science on traveling with a laptop computer. I found it interesting, amusing and easy to relate to. At my wife's request I have been doing some work on address labels. I have worked up a program that allows you to use TI Writer to print anything you want on a label. I hope to demo it at a future meeting. With genealogy demos at the next meeting, it may be a while before we get to it. An excellent article in one of our exchange newsletters (Which one I unfortunately have forgotten.) states that a distance of one inch will usually keep your disks safe from damage. Remembering Murphys law number three that states: The impossible will often happen but only at the worst possible time and with the worst possible results. I keep mine two feet away.

EOF..Jack Johns

Article 994 of comp.sys.ti:
Newsgroups: comp.sys.ti
Path: zodiac.cca.cr.rockwell.com!moe.ksu.ksu.edu!ux1.cso.uiuc
cis.umn.edu!daven
From: daven@vx.cis.umn.edu (David Nieters)
Subject: V9938 Graphics G4 mode tutorial Part 3
Message-ID: <16APR199316571406@vx.cis.umn.edu>
News-Software: VAX/VMS VNEWS 1.41
Sender: daven Nntp-Posting-Host: vx.cis.umn.edu

Now - on with part 3.

In part 2, we used a command built into the V9938 video chip for drawing dots on the screen. In this part, we will use another command that will draw the whole line for us. But first, we must learn about a couple of more registers.

In part two, I made a reference to a status bit that will have to be checked before giving the V9938 a command to execute. The reason we weren't checking this bit in part two is that we were using such a simple command that the 9938 would finish executing it before we tried to give it another command. The line drawing command (that we will use in this part) takes more time to execute and the possibility arises where the 9900 will compute the endpoints for the next line before the 9938 drew the last one.

The bit that needs to be checked is the Command Execute Flag (CE) in bit 0 of status register #2. The 9918(A) had only one status register. When you wanted to read its value, you would just read the byte at location VDPSTA. The V9938 has 10 status registers (#0 - #9.) If you want to read the value of any one of them, you still read from the location VDPSTA, but first you must tell the 9938 which one you want to read. You do this by setting the status register number in Register #15. You will see in the code and extra status register and value added to the list (DATA >0F02). This is where I tell the 9938 which status register I want to read. You will notice that I set it back to 0 when I exit. Status register #0 on the 9938 is the same as the status register on the 9918(a).

Now for the line drawing command. It's not as simple as you would like it to be. You would just like to put in th e end points and let the 9938 connect the dots. Unfortunately, that's not how it works. You have to give the 9938 one end point (X,Y) coordinate. You then have to tell it that the line will either move to the left or to the right, and whether it will move up or down. You then tell it how many dots to move in one direction and how many dots to move in the other direction. Now you may be thinking, "OK, I tell it where to start, and how many dots to go in the X-direction and how many dots to go in the Y-direction, and then we go left, right, up or down - easy." Well, it's still not that easy because you don't give an X-distance and a Y distance. You must determine which way is the longest and tell the 9938 if that is the X direction or the Y direction.

You're probably getting confused, so let's walk through an example. Let's say we wanted to draw a line from the top left hand corner of the screen (1,1) to the bottom right hand corner of the screen (255,191.) We can use either point as our starting point, so I will use (1,1.) We put that in registers R#36-R#39 just like setting the coordindate of a point in part 2. To refresh your memory, R36 is the lower 8 bits of the X coordindate. R#37 has the high order bit of the X coordinate. R#38 has the lower 8 bits of the Y coordindate. R#39 has the high 2 bits of the Y coordinate.

We are going 255 dots in the X direction and 192 dots in the Y direction. 255 is the larger number, so that gets stored in R#40. 192 is the smaller number so that gets stored in register #42. If either number were larger than 255, the high order bits would be stored in R#41 and R#43 respectively. We store the color we want the line to be drawn in R#44. Since G4 mode has only 16 colors, the lower 4 bits of R#44 are set to the color, and the higher 4 bits are set to zero.

R#45 has a few bits that need to be expl ained. It looks like this:

```
----------------------------------------

| 0 | - | 0 | - | DIY | DIX | - | MAJ | <- R#45

----------------------------------------

| 0 | 0 | 0 | 0 |  1  |  0  | 0 |  0  | <- What we will store

----------------------------------------
```

The DIY bit tells the 9938 if we want the line to move to up or down. A value of 0 indicates the line should go up. A value of 1 indicates the line should go down. The DIX bit tells the 9938 if the line should go to the left or to the right. A value of 0 indicates the line should go to the right. A value of 1 indicates the line should go to the left. Finally, the MAJ bit tells the 9938 which coordinate goes furthest. A value of 0 indicates that the X direction is longer . A value of 1 indicates that the Y direction is longer or equal to the X direction.

Once we have all the registers set, we tell the 9938 to execute the command by setting register #46. To make it draw a line, we store a value of 70 (Hex.) So here is how we set the registers to draw this line -

```
R#36 = 1 \ X coordinate of endp oint
R#37 = 0 /
R#38 = 1 \ Y coordinate of endp oint
R#39 = 0 /
R#40 = 255 \ dX
R#41 = 0 /
R#42 = 191 \ dY
R#43 = 0 /
R#44 = 15 Color = white
R#45 = 8
R#46 = >70
```

The code that follows does that same line drawing that parts 1 and 2 did, but now we use the line drawing command of the 9938. We have to do a little extra work to compute which direction is further, and if we go left, right, up, or down, but we no longer have to call the POINT subroutine to draw each individual dot. You will see an amazing speedup over part 2. Also, try taking out the check for the Command Execute Flag and see what happens. Some of the longer lines will be broken. That is because you are writing new values in the line registers before the line had finished being drawn.

```
       REF  VWTR,VSBW,VMBW,KSCAN,VSBR
       REF  VDPWD,VDPWA,VDPSTA
HEIGHT EQU  212          NUMBER OF LINES
NUMLIN EQU  500          NUMBER OF LINES WE DRAW BEFORE ERASING SCREEN
* CLEAR THE SCREEN
*
* THIS ROUTINE CLEARS THE SCREEN BY WRIT ING ZEROS IN THE
* PATTERN NAME TABLE.  WHEN DEALING WITH THE LARGER MEMORY
* SPACE OF THE V9938, WE HAVE TO BE SURE THAT REGISTER #14
* IS CLEARED BEFORE WE START.  OTHERWISE WE MIGHT BY ZEROING
* OUT HIGHER AREAS OF MEMORY THAN WE WAN T TO.
*
CLEAR  LI   R0,>0E00     RESET OUR VDP ADDRESS
       BLWP @VWTR
       LI   R0,>0040
       MOVB R0,@VDPWA
       SWPB R0
       MOVB R0,@VDPWA
       LI   R2,HEIGHT8
CLEAR1 CLR  @VDPWD
       DEC  R2
       JNE  CLEAR1
       LI   R0,>0E00     RESET IT AGAIN SO WHEN WE HIT QUIT
BLWP @VWTR       THE TITLE SCREEN IS OK
RT
* RANDOM NUMBER GENERATOR
*
* THIS PROCEDURE RETURNS A (NOT SO) RAND OM NUMBER IN R1.
* IT ENSURES THE RANDOM NUMBER WILL NOT BE 0.
*
RAND   MOV  @SEED,R1
RAND1  AI   R1,>1D6B
       JEQ  RAND1
       MOV  R1,@SEED
       RT
SEED   DATA >690A
DX1    DATA 0           THESE LOCATIONS ARE USED TO STORE
DX2    DATA 0           HOW FAR THE ENDPOINTS MOVE EACH
DY1    DATA 0           TIME A LINE IS DRAWN
DY2    DATA 0
* COLOR FLAG
*
* WHEN COLOR FLAG IS ZERO, THE LINES WIL L APPEAR IN
* DIFFERENT COLORS.  WHEN IT IS NOT SET TO ZERO, ALL
* LINES WILL BR DRAWN IN THE SAME COLOR.  IT'S TOGGLED
* BY PRESSING THE 'C' WHILE LINES ARE BE ING DRAWN.
```

```
*
CFLAG  DATA 0
* PLOT
*
* THIS ROUTINE PLOTS A LINE FROM (X1,Y1) TO (X2,Y2)
* THESE COORDINATES ARE LOCATED IN THE C ALLERS
* REGISTERS R6,R7,R8 AND R9.  THE COLOR IS
* SPECIFIED IN THE CALLER'S R10.
*
PLOT   DATA >8300
       DATA PLOT1
PLOT1
       CLR  R5           THIS WILL BE R45
       MOV  @16(R13),R7  X2
       MOV  @12(R13),R9  X1
       S    R9,R7        R7 = dX
       JGT  PLOT11
       ABS  R7           WE WANT dX TO BE A POSITIVE NUMBER
       ORI  R5,>0004     WE ARE GOING LEFT
PLOT11 MOV  @18(R13),R8  Y2
       MOV  @14(R13),R10 Y1
       S    R10,R8       R8 = dY
       JGT  PLOT4
       ORI  R5,>0008     WE ARE GOING UP
       ABS  R8
* HERE WE CHECK THE COMMAND EXECUTE FLAG AND
* MAKE SURE IT'S ZERO BEFORE WE GO ANY F URTHER
*
PLOT4
       MOVB @VDPSTA,R1  MAKE SURE WE CAN GO
       ANDI R1,>0100
       JNE  PLOT4
       MOV  R9,R0        PUT IN X1
       AI   R0,366
       BLWP @VWTR
       LI   R0,376
       BLWP @VWTR
       MOV  R10,R0       PUT IN Y1
       AI   R0,386
       BLWP @VWTR
       LI   R0,396
       BLWP @VWTR
       C    R7,R8        SEE WHICH WAY IS LONGER
       JLT  PLOT5
       MOV  R7,R0        X IS GREATER
       AI   R0,406    SET R#40 TO THE LONGER DISTANCE
       BLWP @VWTR
       MOV  R8,R0
       AI   R0,426    SET R#42 TO THE SHORTER DISTANCE
       BLWP @VWTR
       JMP  PLOT6
PLOT5  ORI  R5,>0001     Y IS GREATER
       MOV  R8,R0
       AI   R0,406    SET R#40 TO THE LONGER DISTANCE
       BLWP @VWTR
       MOV  R7,R0
       AI   R0,426    SET R#42 TO THE SHORTER DISTANCE
       BLWP @VWTR
```

```
PLOT6  LI   R0,416    CLEAR R#41
       BLWP @VWTR
       LI   R0,436    CLEAR R#43
       BLWP @VWTR
       MOV  @10(R13),R0  GET COLOR
       AI   R0,446    STORE COLOR IN R#44
       BLWP @VWTR
       MOV  R5,R0
       AI   R0,456
       BLWP @VWTR
       LI   R0,466+>70 NOW EXECUTE THE COMMAND
       BLWP @VWTR
       RTWP
START  LWPI >8320
       LI   R2,VDPREG    SET VDP REGISTERS
L1     MOV  *R2+,R0
       JLT  L2
       BLWP @VWTR
       JMP  L1
L2     BL   @CLEAR
       CLR  R3           R3 COUNTS THE NUMBER OF LINES WE HAVE DRAWN
       LI   R6,>80       SET THE ENDPOINTS FOR OUR FIRST LINE
       LI   R7,>60
       LI   R8,>D3
       LI   R9,>13
       CLR  R0           SET THE INITIAL AMOUNTS THE END POINTS
       INCT R0           MOVE BY
       MOV  R0,@DX1
       INCT R0
       MOV  R0,@DY1
       INCT R0
       MOV  R0,@DX2
       INCT R0
       MOV  R0,@DY2
LOOP   MOV  @CFLAG,R0
       JNE  L5
       BL   @RAND        PICK A RANDOM COLOR
       ANDI R1,>F
       MOV  R1,R5
       CI   R5,2         MAKE SURE WE DON'T HAVE A BLACK
       JHE  L5
       ORI  R5,2
L5     A    @DX1,R6      MOVE THE ENDPOINTS
       A    @DY1,R7
       A    @DX2,R8
       A    @DY2,R9
*·CHECK TO MAKE SURE THAT NO ENDPOINTS H AVE MOVED OFF
* THE SCREEN.  IF SO, REVERSE ITS DIRECT ION.
*
       MOV  R6,R6
       JLT  L6
       CI   R6,>100
       JLT  L7
L6     NEG  @DX1
       A    @DX1,R6
L7     MOV  R8,R8
       JLT  L8
       CI   R8,>100
       JLT  L9
L8     NEG  @DX2
       A    @DX2,R8
L9     MOV  R7,R7
       JLT  L10
       CI   R7,HEIGHT
       JLT  L11
L10    NEG  @DY1
       A    @DY1,R7
L11    MOV  R9,R9
       JLT  L12
       CI   R9,HEIGHT
       JLT  L13
L12    NEG  @DY2
       A    @DY2,R9
L13    BLWP @PLOT
L14    CLR  R0           CHECK TO SEE IF A KEY IS PRESSED
       MOVB R0,@>8374
       BLWP @KSCAN
       MOVB @>8375,R0
       MOVB @>837C,R1
       JEQ  L16
       CI   R0,>0500     CHECK FOR QUIT KEY
       JNE  L15
       B    @QUIT
L15    CI   R0,>4300     CHECK FOR "C" KEY PRESSED
       JNE  L14
       INV  @CFLAG       TOGGLE THE COLOR FLAG
L16    CI   R0,>FF00
       JNE  L14
       INC  R3
       CI   R3,NUMLIN    SEE IF WE HAVE MORE LINES TO DRAW
       JNE  LOOP         IF SO, GO BACK AND DRAW THEM
       CLR  R3
       LI   R2,6
       CLR  R4
DLY    DEC  R4           WAIT A LITTLE BEFORE CLEARING THE SCREEN
       JNE  DLY
       DEC  R2
       JNE  DLY
       BL   @RAND        COMPUTE NEW RANDOM MOVEMENTS
       MOV  R1,R1
       JLT  L17
       ANDI R1,7
       JMP  L18
L17    ORI  R1,>FFF8
L18    MOV  R1,@DX2
       BL   @S7CF2
       MOV  R1,R1
       JLT  L19
       ANDI R1,7
       JMP  L20
L19    ORI  R1,>FFF8
L20    MOV  R1,@DY1
       BL   @S7CF2
       MOV  R1,R1
       JLT  L21
       ANDI R1,7
       JMP  L22
L21    ORI  R1,>FFF8
L22    MOV  R1,@DX1
       BL   @S7CF2
       MOV  R1,R1
       JLT  L23
       ANDI R1,7
       JMP  L24
L23    ORI  R1,>FFF8
L24    MOV  R1,@DY2
       BL   @CLEAR        CLEAR SCREEN
       B    @LOOP         START OVER
QUIT   LI   R2,REG2       RESTORE VDP REGISTERS BACK TO NORMAL
QUIT1  MOV  *R2+,R0
       JLT  QUIT2
       BLWP @VWTR
       JMP  QUIT1
QUIT2
       LIMI 2
       BLWP @0
* VDP REGISTERS TO SET VDP TO GRAPHICS 4 MODE
*
VDPREG DATA >0006
       DATA >0160
       DATA >021F
       DATA >03FF
       DATA >0403
       DATA >0536
       DATA >0717
       DATA >080A         INHIBIT SPRITES
       DATA >0988         212 LINES, INTERLACE
       DATA >0F02         SO WE CAN READ STATUS REGISTER #2
       DATA >FFFF

REG2   DATA >0000
       DATA >0F00
       DATA >01F0
       DATA >0200
       DATA >03FF
       DATA >0401
       DATA >0560
       DATA >0F00         READ STATUS REGISTER #0 AGAIN
       DATA >FFFF

       END  START
```

What is electricity?

Mike Boyd, N7DKQ

Today's scientific question is: What in the world is electricity? And where does it go when it leaves the toaster?

Here is a simple experiment that will teach you an important electrical lesson. On a cool, dry day, scuff your feet along a carpet, then reach your hand into a friend's mouth and touch one of his dental fillings. Did you notice how your friend twitched violently and cried out in pain? This teaches us that electricity can be a very powerful force, but we must never use it to hurt others unless they need to learn an important electrical lesson.

It also teaches us how an electrical circuit works. When you scuffed your feet, you picked up a batch of ;*(' @ electrons which are very small objects that carpet manufacturers weave into carpets so they will attract dirt. The electrons travel through your bloodstream and collect in your finger, where they form a spark that leaps to your friend's filling. It then travels down to his feet and back into the carpet, thus completing the circuit.

Amazing electronic fact: If you scuffed your feet long enough without touching anything, you would build up enough electrons to explode your finger! But this is nothing to worry about unless you have carpets. Although we modern people tend to take our electric lights, radios, mixers, etc. for granted, hundreds of years ago people did not have any of these things, which is just as well because there was no place to plug them in.

Then along came the first electrical pioneer, Ben Franklin, who flew a kite in a lightning storm and received a serious electrical shock. This proved that lightning was powered by the same force as carpets, but it also damaged Franklin's brain so severely that he started speaking in incomprehensible maxims such as "A penny saved is a penny earned" and more. Eventually he had to be given a job running the post office.

After Franklin came a herd of electrical pioneers whose names have become part of our electrical terminology. Myron Volt, Mary Louise Amp, James Watt, Wayne Green and Bob Transformer are a few. These pioneers conducted many important electrical experiments. For example, in 1780 Luigi Galvani discovered (this is true) that when he attached two different kinds of metal to the leg of a frog, an electrical current developed and the frog's leg kicked, even though it was no longer attached to the frog, which was dead anyway. Galvani's discovery led to enormous advances in the field of amphibian medicine. Today, skilled veterinary surgeons can take a frog that has been seriously injured or killed, implant pieces of metal in its muscles and watch it hop back into the pond just like a normal frog, except for the fact that it sinks like a stone.

The greatest electrical pioneer of all was Thomas Edison, who was a brilliant inventor despite the fact that he had little formal education and lived in New Jersey. Edison's first major invention, in 1877, was the phonograph, which could soon be found in thousands of American homes, where it basically sat until 1923 when the record was invented. Edison's greatest achievement came in 1879, when he invented the electric company. Edison's design was a brilliant adaptation of the simple electrical circuit: The electric company sends electricity through a wire to a customer, then immediately gets the electricity back through another wire, then (this is the brilliant part) sends it right back to the customer again.

This means that an electric company can sell a customer the same batch of electricity thousands of times a day and never get caught, since very few customers take the time to examine their electricity closely. In fact, the last year in which any new electricity was generated in the United States was 1937. The electric companies have been merely reselling it ever since, which is why they have so much free time to apply for rate increases.

Today, thanks to men like Edison and Franklin, and frogs like Galvani's, we receive almost unlimited benefits from electricity. For example, in the past decade, scientists developed the laser, an electronic appliance that emits a beam of light so powerful that it can vaporize a bulldozer 2,000 yards away, yet so precise that doctors can use it to perform delicate operations on the human eyeball, so long as they remember to change the power setting from VAPORIZE BULLDOZER to DELICATE OPERATION.

— Oregon Tualatin Valley ARC, Beaverton, OR

CEDAR VALLEY 99er UG   OCTOBER 1993

NEXT MEETING: Tuesday

October 12, 1993 6:30PM

WEST MUSIC, COLLINS ROAD
PLAZA, MARION
ACROSS FROM LINDALE MALL

Cedar Valley 99'er Users Group
c/o Jim Green
377 Cambridge Dr. NE
Cedar Rapids, Iowa 52402-1446

FIRST CLASS

Send To:

GARY BISHOP
124-222
3270 28TH AVE
MARION, IA 52302