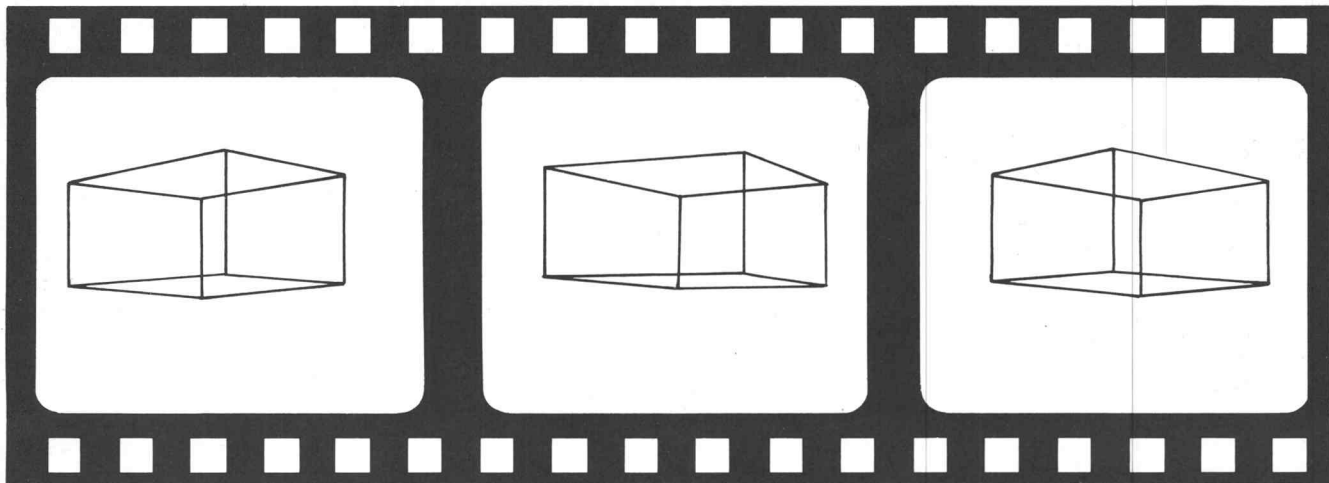


Tekniques

a Tektronix applications newsletter published monthly

Vol. 1 No. 5



Computer Animation With The 4051

by Paul Doherty, Oakland University, Rochester, Michigan

Inexpensive computer animated films can be produced by combining a super-8 motion picture camera and a TEKTRONIX 4051 graphics system. The technique of computer animation extends the capabilities of the 4051. For example, several hours of calculating and drawing the time dependence of complex data plots, can be compressed to several minutes of film. This adds a third dimension, time, to the graphics display system. The study of the time dependence of data is an important part of my use of the graphics display system, teaching undergraduate physics. Computer animation is also useful in bringing out the three spacial dimensions of an object. If the computer draws successive perspective views of a 3-D object or data plot from a continuously moving point of view, the human eye-brain computer will then quite easily perceive the 3-D nature of the object. This same computerized tour can be performed with successive stereo pairs. There are many other possibilities.

In computer animation the computer draws one frame of a movie at a time. After each frame, it sends a command through the GPIB (General Purpose Interface Bus), to a camera, which exposes one frame of film.

Very few movie cameras are suitable for the production of computer animated films on the 4051. An XL-camera is required to handle the low light level of the display. It must retain high resolution, have single frame capability, macro-focusing (preferably at wide-angle focal lengths), and an electromagnetic shutter release.¹ Let's discuss each requirement in more detail.

¹We use a Chinon 555X1, super-8 camera, which is available for about \$210.

Exposure

Photographing the 4051 with a super-8 camera requires the use of an $f/1.2$ lens, an XL shutter, which exposes film for 1/28 second, and an ASA 160 film. The above requirements push the current state of the art for super-8 cameras to the limit. The resolution of the resulting image is intimately connected with exposure.

(Continued on page 6.)

Speed Up Nonlinear Least Squares With Function Minimization

by Gary P. Laroff

(This is the second in a series of articles on applications of the new Tektronix software package 4050A10 Statistics Volume 4.)

In the last issue of Tekniques (Volume 1, Number 4) we discussed nonlinear regression analysis using Marquardt's Least Squares method (program 1 in STAT 4). Nonlinear least square methods, either Marquardt's or

(Continued on page 4.)

In This Issue

New Software Ad Service	Page 2
Program Tips	Page 2
Recording and Displaying TSC Graphics	Page 3
New Abstracts	Page 7

Using The 4051 As a Classroom Aid

by Mary Jo Feuerstein

Have you ever considered the possibility of using your 4051 as a teaching aid? Many of the members of the 4051 Applications Library are using their 4051s for high school and college-level educational aids both as individual tutorial stations and as a classroom aid to complement lectures. Mr. Paul Doherty of Oakland University in Rochester, Michigan, for example, has been using his 4051 to create animated, super-8 films of physical and scientific phenomena. (See Mr. Doherty's article on page 1 of this issue of *Tekniques*.)

Others of you have developed BASIC programs on your 4051s to use in courses ranging from geometry to chemistry, health science to music, and electrical engineering to the social sciences. You might use the 4051 to graphically display the growing marketplace of company A in your college-level marketing class, allow a student in your physics class to delve more deeply into the theory of gravitational acceleration on his own, or bring a 4051 to a calculus class and use it for graphic analysis of the principles of integration, while you explain the mathematics.

If you have been using your 4051 in an education environment, the 4051 Applications Library and the other library members would like an opportunity to share your educational resources. You can do this either by submitting one or more of your education programs to the library or write us a note explaining how you are using your 4051 to make your teaching job easier. We'd also like to hear about any needs or ideas you have for education programs.

Through the 4051 Applications Library, Tektronix can act as a focal point for input and distribution of 4051 educational tools. Remember, if you have a program included in the 4051 Applications Library, you can have your choice of any three programs from the existing library. Send us one educational tool and get three in return.

Tekniques, the 4051 Applications Library Newsletter, is published monthly by the Information Display Group of Tektronix, Inc. It is distributed to TEKTRONIX 4051 users and members of the 4051 Application Library.

**Publisher
Managing Editor
Editor**

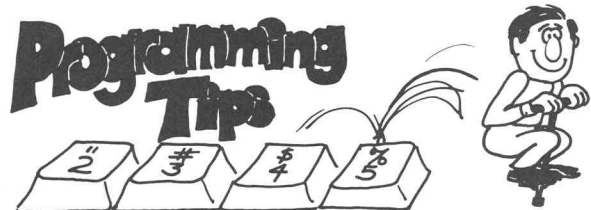
**Ken Cramer
Sharon Simmons
David Schwabe**

Volume 1, Number 5. Copyright 1977 Tektronix, Inc. Printed in the United States of America. All Rights Reserved.

Tekniques Offers New Software Ad Service

In our next issue, *Tekniques* will be offering a new feature, Software Notices. A Software Notice, like a classified ad, will allow those of you who have developed 4051 software that you wish to sell, to notify other 4051 owners and users. Each Software Notice will contain the program or system name, an abstract or description, the technical requirements, the name of the person, company, or organization offering the program, the address and telephone number, and the price if desired.

You can obtain Software Notice Submittal forms by writing to Tektronix, P.O. Box 500, Beaverton, Oregon 97077, c/o Applications Library 60-369.



Pointing To Areas On The Screen

by Ken Cramer

Do you ever need the ability to point to locations on the 4051 screen for an X,Y readout, or to indicate a corner for a window or viewport? The 4952 Option 2 Joystick and the 4051 POINTER command are the most efficient way to do this, requiring only one line of BASIC code. If you don't have a 4952, however, similar kinds of functions can be performed with the User Definable Keys and the following subroutine:

```
1 GO TO 100
2 REM THE CURSOR MOVEMENT IS CONTROLLED
3 REM BY THE FUNCTION KEYS SIX THRU TEN
24 REM DECREMENT THE X VALUE
25 X9=X9-J9
26 RETURN
28 REM INCREMENT THE X VALUE
29 X9=X9+J9
30 RETURN
32 REM DECREMENT THE Y VALUE
33 Y9=Y9-J8
34 RETURN
36 REM INCREMENT THE Y VALUE
37 Y9=Y9+J8
38 RETURN
40 REM REGISTER THE POINT AND
41 REM RETURN TO THE MAIN PROGRAM
42 J7=1
43 RETURN
83 REM STATEMENT 85 IS THE ENTRY POINT
84 REM FOR THE SUBROUTINE
85 SET KEY
86 J9=(M7-M6)/200
87 J8=(M9-M8)/200
88 X9=M5+(M7-M6)/2
89 Y9=M8+(M9-M8)/2
90 PRINT #32,18:5
91 J7=0
92 MOVE X9,Y9
93 PRINT #32,24:"1"
94 IF J7=0 THEN 92
95 PRINT #32,18:0
96 RETURN
```

With this subroutine you can move the "pointer" around on the 4051 screen by pressing User Definable Keys 6 (left), 7 (right), 8 (down) and 9 (up). Variables J8 and J9 in the subroutine set the horizontal and vertical increments, respectively, that the pointer is moved each time one of the User Definable Keys is pressed. Press key

(Continued on next page.)

10 to exit from the subroutine and return the X,Y location of the pointer to the main program. X and Y are returned to variables X9 and Y9 in the main program, in user coordinates defined by the current WINDOW definition parameters W6, W7, W8 and W9.

GOSUB 90 invokes the subroutine if the current WINDOW definition is acceptable. Enter the subroutine at 85 if a new WINDOW definition has been defined or to position the pointer to the center of the crt. The following is a sample program that calls this pointer positioning subroutine:

```

100 REM THE MAIN PROGRAM WOULD START AT LINE 100
110 REM AND REFERENCE THE LOCATOR SUBROUTINE BY
120 REM EXECUTING A "GOSUB 85"
130 REM
140 REM THIS IS A SIMPLE PROGRAM WHICH PRINTS THE X,Y
150 REM VALUES ON THE SCREEN
160 W6=-200
170 W7=200
180 W8=0
190 W9=150
200 WINDOW W6,W7,W8,W9
210 GOSUB 85
220 PRINT X9;"",Y9
230 GOSUB 90
240 GO TO 220

```

Variations on this theme can be obtained using different values for J8 and J9, and using additional User Definable Keys for coarse positioning.

Plot Arrays Faster and More Efficiently

A typical method of plotting arrays of X,Y points is to use a FOR-NEXT loop, where N equals the dimension of the X and Y arrays.

```

200 MOVE X(1),Y(1)
210 FOR I=1 TO N
220 DRAW X(I),Y(I)
230 NEXT I

```

You can create this display faster and with more efficient coding if you use the built in array capability of 4051 BASIC. The same display can be obtained with the following shorter code:

```

200 MOVE X(1),Y(1)
210 DRAW X,Y

```

For more information on the array output capability of 4051 BASIC see page 9-24 in the 4051 Graphic System Reference Manual.

Recording And Displaying TCS Graphics On The 4051 Graphic System

by Chuck Eng

The 4051 is a stand-alone graphic system, but it also has the ability to receive and display graphic data from an outside source (such as a computer mainframe), as if it were a graphic display terminal. Tektronix' graphic display terminals like the 4010 Series use the Terminal Control System (TCS) Plot 10 software to perform

graphic operations. A 4051 with an Option 1 data communication interface can perform these same graphic display functions using the Option 1 4012 Emulation mode. In this terminal mode, the 4051 receives TCS Plot 10 graphic data and displays it directly on the screen (see Fig. 1). If your 4051 is equipped with Option 1, the RS232 interface, you can also store this TCS Plot 10 graphic data on the 4051 internal magnetic tape, using the DATA RECEIVE mode. See the 4051 Option 1 Manual for more information on storing data received through the RS232 interface on tape.

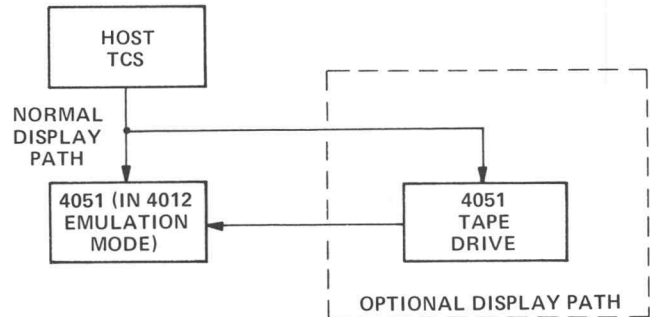


Fig. 1. Graphic data can be displayed directly on the 4051 screen or stored first on tape and then transferred to the screen.

This recording and display procedure is accomplished without using any software to do the data conversion or handle the graphics. Once the graphic data is stored on the tape, use the following procedure to transfer the data from the tape to the screen:

1. Connect the Self Test Adapter (Tektronix Part No. 013-0173-00) to J93, the RS232 port, on the rear panel of the 4051, or the Self Test Adapter (Tektronix Part No. 013-0173-01) to the interconnect cable.
2. CALL "RATE" to reset the baud rate. 2400 baud is recommended.
3. CALL "TERMIN".
4. Press User Definable Key 16, ECHO OFF, to turn the local echo off.
5. Find the beginning of the graphic data file.
6. Press User Definable Key 4, DATA SEND.

If your TCS graphics data already exists on a 4923 Digital Cartridge Tape, this procedure should be preceded by the following procedure:

1. Place the 4923 tape in the 4051 internal tape drive.
2. Type Pri @ 33,0: 1,1,1 to set the 4051 in the "No-Header" mode.
3. Continue with the above procedure.

Functional Minimization (continued from page 1.)

the Gauss-Newton method (program 2 in STAT 4), use an iterative procedure to converge on the best values for the unknown parameters, starting with initial estimates of their values provided by the program user. Computation time for either of these methods is dependent on the complexity of the assumed model and the number of sets of data values. The initial estimates of the parameters of the function also play a crucial role in determining how long it will take for the program to converge. Some choices for initial parameters are so far from the actual values that the computation either requires an excessive amount of time or simply doesn't converge. You can frequently make reasonable estimates, but this is not always the case.

The best initial estimates are often those that produce the smallest value of the "Initial Sum Of Squares", which is printed at the beginning of the calculations. An empirical approach would be to try a number of sets of estimates, stopping the program manually after it prints the initial sum of squares. The initial estimates producing the smallest sum could then be used for the computation. A better approach is to let a program iteratively determine this sum for you.

Nelder and Mead Function Minimization

Nelder and Mead Function Minimization, program 3 in the STAT 4 package, uses the Nelder-Mead simplex procedure to find the minimum of a known function. The function must be continuous, and a local minimum must exist. With a short user written subroutine, this program can be used to optimize the input parameters for the nonlinear least squares programs.

Before we discuss this optimization application, let's look at a simple example of how Nelder and Mead Function Minimization can be used to find the local minimum of a function. Fig. 1 shows a parabola defined by:

$$Y = -11 + 3(X + 4)^2$$

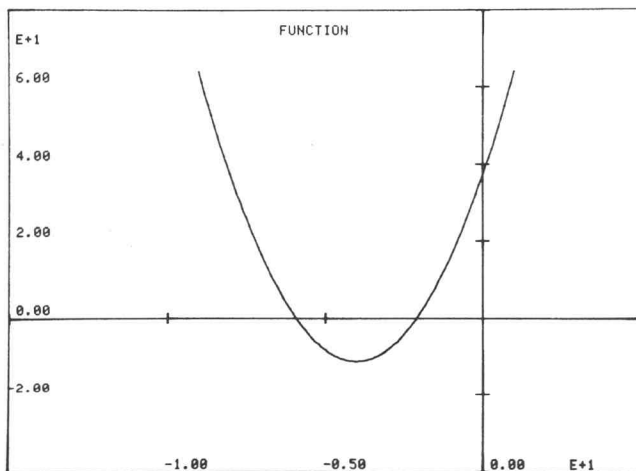


Fig. 1. Graph of parabola with minimum of -11 at X = -4.

This function is keyed into program 3 as:

$$Y = f(X(I))$$

where X(I) represents the independent variables X(1), X(2), ..., X(n). The actual equation for the parabola is accessed through a short user written subroutine, which begins at line 1000.

```
1000 Y=-11+3*(X(1)+4)^2
1010 RETURN
```

With an initial estimate of the minimum at X = -1 (an absurd guess) and a requested criteria of convergence of less than 1.0 E-10, the function minimum of -11 is found at X = -4 in 30 iterations (35 seconds, printing each iteration). Better initial estimates would decrease both the number of iterations and the computation time, considerably.

Function Minimization to Optimize Nonlinear Least Squares

Both the Marquardt and Gauss-Newton methods of nonlinear least squares analysis minimize the sum of the least squares to determine the parameters of the selected function. That is, they calculate the difference between the Y-value of an actual data point and the Y-value obtained from the function (for an identical X-value), square the difference, and calculate the sum of the squares obtained for each data point. This sum is then minimized over a number of iterations and changes of parameters, until a specified minimum variance in the sum of the squares is reached.

This same procedure can be implemented with Nelder and Mead Function Minimization by including the data pairs, the proposed function, and a least squares function in the user written subroutine. The subroutine reads the data pairs from a tape and calculates Y-values from the proposed function and parameters. Program 3 then proposes new parameters for the proposed functional model, using the Nelder-Mead simplex method, to minimize the sum of the squares function.

Nelder and Mead Function Minimization has an advantage over Marquardt's or Gauss-Newton nonlinear least squares methods in that Nelder and Mead will converge from any choice of initial parameters. Nelder and Mead is not as fast, however, as Marquardt's or Gauss-Newton. Nelder and Mead is thus used to optimize (determine ball park estimates for) the initial parameters to be used in the Marquardt's or Gauss-Newton nonlinear least squares programs. The Marquardt's or Gauss-Newton programs will then converge very quickly.

The following subroutine is used to optimize the parameters for the function used in last month's article on Marquardt's nonlinear least squares method. This sub-

routine is general purpose; it can be used to optimize the initial parameter for the function Y2, which can then be solved with either of the nonlinear least squares programs, one or two.

```

500 REM *** LINES 500-630 READ IN YOUR DATA FROM TAPE ***
510 PRINT "ENTER NUMBER FOR TAPE FILE CONTAINING DATA: ";
520 INPUT N9
530 PRINT "INSERT DATA TAPE AND PRESS "RETURN""
540 INPUT N9
550 FIND N9
560 READ @33:N9
570 DELETE X1:Y1
580 DIM X1(N9),Y1(N9)
590 FOR I=1 TO N9
600 READ @33:X1(I),Y1(I)
610 NEXT I
620 RETURN
630 REM *** THIS IS THE END OF THE ROUTINE THAT READS IN YOUR DATA ***

1000 Y=0
1010 FOR I=1 TO N9
1020 REM *** YOUR FUNCTION GOES NEXT ***
1030 Y2=X(1)*X(3)/(X(1)-X(2))*(EXP(-X(2)*X1(I))-EXP(-X(1)*X1(I)))
1460 REM *** YOUR FUNCTION HAS TO FINISH BY THIS POINT ***
1470 Y=Y+(Y1(I)-Y2)^2
1480 NEXT I
1490 RETURN

```

Lines 500 through 900 are executed only once. Lines 500-550 request the number of the tape file on which the data is stored and reminds the operator to insert his data tape. Lines 560 to 630 read the X,Y data pairs into the subroutine so that minimization can be computed on the operator's data. Line 560 reads in the number of data pairs as variable N9. Lines 580-610 read in the data as arrays X1 (N9), Y1 (N9). This is the standard data format for data storage as used in the nonlinear regression programs.

Lines 1000-1490 are executed for every iteration of the function minimization. Lines 1010-1480 loop through the user function for each of the data points.

Line 1030 contains the proposed function model, in this case:

$$Y = \frac{C_1 * C_3}{C_1 - C_2} * (e^{-C_2 X} - e^{-C_1 X})$$

where $Y2 = Y$
 $X1(I) = X$
 $X(1) = C_1$
 $X(2) = C_2$
 $X(3) = C_3$

The initial parameters that would be entered in the nonlinear regression program are entered for X(1) through X(3).

Remember, this is not the function being minimized. It is only used to calculate the Y-values used in the sum of the squares function:

$$Y = Y + (Y1(I) - Y2)^2$$

where $Y1(I) = Y$ -value of data point $X1(I)$
 $Y2 = Y$ -value of proposed function model for $X = X1(I)$ and initial parameters $X(1)$ through $X(3)$

Line 1470 computes the sum of the squares and serves as the function to be minimized.

Pressing User Definable Key 1 initiates the program. The number of variables, three in this case, is then entered, and the starting values are then requested, as in Fig. 2. The values entered here are the same ones used in the example last month. The new subroutine then requests convergence criteria, the data file number, insertion of the data tape and computes the best values of the initial parameters. The "Best Answers" from the Function Minimization program should be recorded and used as the initial parameters for the same data set and function when used with the Marquardt's or Gauss-Newton Nonlinear Least Squares programs. These new parameters are much closer to the final parameters calculated by the least squares programs than the initial estimates. Using these optimized parameters, the least squares program will run faster and more efficiently.

```

ENTER STARTING VALUES
STARTING VALUE FOR X(1) = 1
STARTING VALUE FOR X(2) = 0.1
STARTING VALUE FOR X(3) = 100

THE CRITERIA FOR CONVERGENCE IS THE VARIANCE <= 1.0E-6
DO YOU WISH TO CHANGE THIS VALUE ? NO

CONVERGENCE IS TESTED EVERY 1 ITERATIONS.
DO YOU WISH TO CHANGE THIS FREQUENCY ? NO

RESULTS ARE PRINTED EVERY 1 ITERATIONS.
DO YOU WISH TO CHANGE THIS FREQUENCY ? NO

ENTER NUMBER FOR TAPE FILE CONTAINING DATA: 33
INSERT DATA TAPE AND PRESS "RETURN"

```

Fig. 2. Initial printout from Nelder-Mead Functional Minimization program.

Initial Estimates	Best Answer From Function Minimization	Final Parameter From Least Squares
1.	1.29973	1.30013
0.1	0.09997	0.09999
100.	111.98469	111.98932

This parameter optimization subroutine can be used to optimize initial parameters with the Nelder and Mead Function Minimization program for any function. Merely change the function in line 1030.

A final note. As has been explained in this article, the Marquardt's and Gauss-Newton nonlinear analysis programs use the least sum of the squares method of determining parameters for the proposed function model. The Nelder and Mead Function Minimization program offers you the ability to determine parameters on a basis other than least squares. For example, you may want to minimize the maximum Y-value residual (mini-max fit). In this case you would substitute the following equation for the sum of the squares equation in line 1470.

$$1470 Y=Y+MAX ABS(Y1(I)-Y2)$$

You might also want to minimize the sum of the absolute values of the Y-value residuals.

$$1470 Y=Y+ABS(Y1(I)-Y2)$$

Next Month: Solutions to nonlinear systems of equations.

Resolution

Tektronix graphic terminals provide high resolution displays. They can resolve several hundred lines per frame. These lines must be recorded on 5 mm, super-8 film. Only one color film, the new Ektachrome SM 7244, has sufficient resolution, 100 lines/mm, and speed ASA 160, to match the 4051.² In black and white, tri-X has similar speed and resolution. (Ektachrome 160 G has unacceptable resolution, 40 lines/mm.) The limiting factor in the resolution of super-8 systems with the above properties is the zoom lens when it is operated at f/2. One camera, the Nizo Pro, overcomes the above difficulties. It allows a manually adjustable exposure time of up to 1 minute per frame. It is priced, however, near \$1000.

Focus

At the large apertures used in computer animation, the depth of field is only a few millimeters. The camera must thus be placed an exact distance from the screen and exactly perpendicular to the screen for each filming session. A telescoping aluminum tube or other jig is a useful aid for positioning the camera.

Focal Length

The camera should be able to fill a super-8 frame with the 4051 screen. It should thus focus on a 6X8 inch screen. This can be accomplished with an 8 mm focal length lens, which can focus to 20 cm. In general, such lenses will be labelled as Macro-focusing. An additional close-up lens can also be added.

²This film is available at about 30 locations nationwide. Write to Kodak for information. In the midwest, film and processing are available at Legal Tapes, 8 Mile Road, Detroit, Mich. It can be processed as VNF video newsfilm by many local television stations.

Trigger

Cameras with electromagnetic shutter releases are the easiest to interface to the 4051. The camera is triggered on command from the GPIB. The circuitry described in Figure 1 assumes there is at least one intelligent device attached to the GPIB that will perform the necessary GPIB handshake. When a command such as "PRINT @13:" is executed it causes a negative pulse to be sent over the ATN line of the GPIB. This pulse is then used to trigger a 555 one-shot, which in turn triggers the camera. It is necessary to address an actual intelligent device on the GPIB since that device must send an appropriate response to the 4051 or an error condition is generated. The circuitry described in Figure 1 is an ATN line sensor, which will trigger the camera when any GPIB command is issued. A 4051 triggered camera produces films without a human attendant and relieves a great deal of the tedium associated with animation.

Note, the 4051 resets the location of the X=0, Y=0 address every time the screen is paged. It does this in an 8 position cycle. Thus, you must page 8 times between frames or speak to your Tektronix service technician about disabling this feature.³

We have already used computer animation to explain tides, to examine objects in 3-dimensions, to display the motions of stars over the next 10⁶ years, and to study the motions of isotherms near an igneous intrusion. How you use it is limited only by your imagination.

³The display "reorigin feature" is incorporated into all 4051's as a protective measure. Disabling this protective feature is not advised, unless you understand fully the possible impact of this modification.

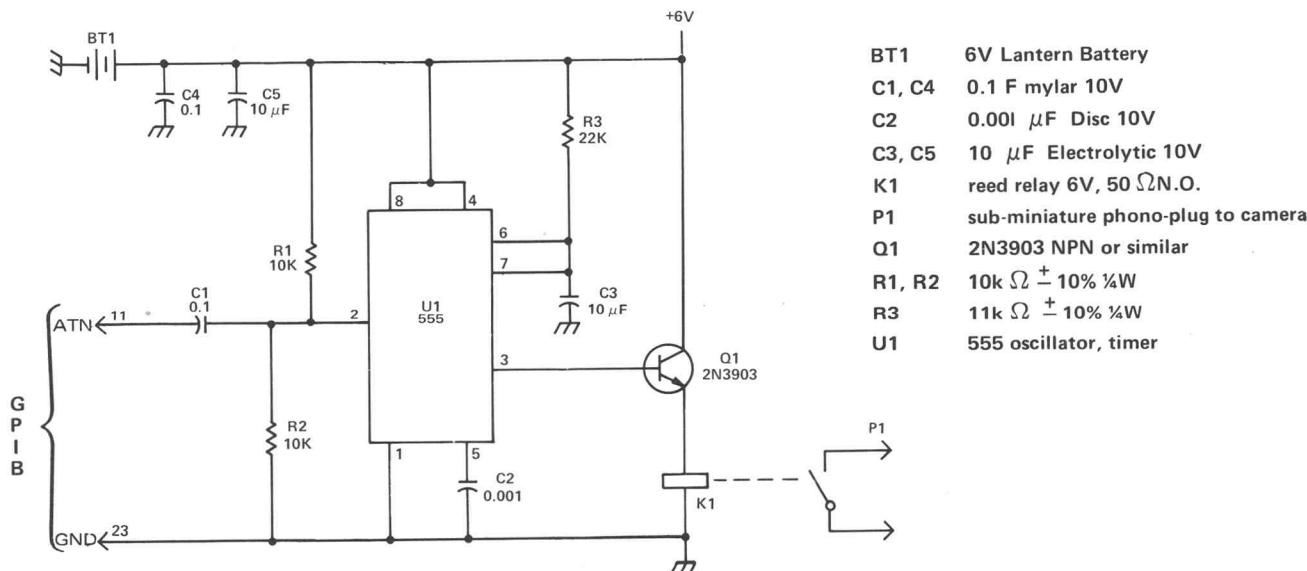


Fig. 1. Camera interface for use with cameras that have electromagnetic shutter release. An attention pulse from the GPIB triggers the one-shot. The relay is then closed for 0.2 seconds and the camera is triggered.

4051 Applications Library Program Abstracts

Documentation and program listings of these programs may be ordered for \$15.00 each. Programs will be put on tape for an additional \$2.00 handling charge per program and a \$26.00 charge for the tape cartridge. (The program material contained herein is supplied without warranty or representation of any kind. Tektronix, Inc. assumes no responsibility and shall have no liability, consequential or otherwise, of any kind arising from the use of this program material or any part thereof.)

Please use the Applications Library Order Form. Order forms are included in the Membership Packet and are available from your local Tektronix Field Office.

● BUSINESS ADMINISTRATION ACCOUNTING

ABSTRACT NUMBER: 51/00-0100/0

Title: **Project Hours Data Maintenance**
Author: Kathy Thurman
Memory Requirements: 24K bytes
Peripherals: Hard Copy, Plotter Optional
Abstract:

This program allows you to maintain data files containing the hours worked per period and the total-to-date hours applied to a project for up to twenty periods. You may summarize the data either by period or compared to the available hours for the period. Both graphic and tabular output are available; graphic output may be either on-screen or through a 4662 plotter.

ABSTRACT NUMBER: 51/00-0101/0

Title: **Engineering Expenses Data Maintenance**
Author: Kathy Thurman
Memory Requirement: 14K bytes
Peripherals: Hard Copy, Plotter Optional
Abstract:

This program allows you to maintain data files containing the expenses accrued per period and the total-to-date engineering expenses applied to a project for up to 50 periods. Both graphic and tabular output are available; graphic output may be either on-screen or through a 4662 plotter.

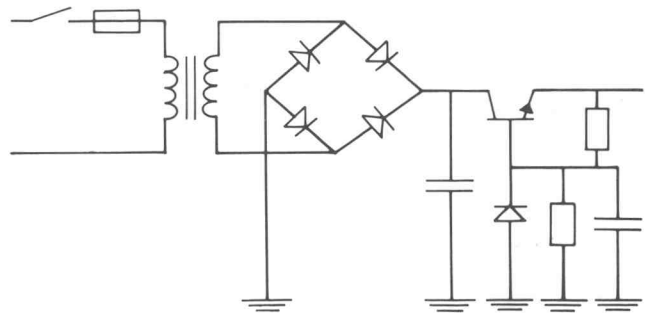
● ENGINEERING ELECTRICAL

ABSTRACT NUMBER: 51/00-1401/0

Title: **Schematic Drawing Program I**
Memory Requirement: 8K bytes
Peripherals: Hard Copy Optional
Abstract:

This program allows you to interactively draw schematics on the 4051 screen. The User Definable Keys are used to select the position of component symbols and connecting lines. A choice of six standard electronic symbols plus a line segment are available: resistor, capacitor, coil, diode, transistor and earth (ground). Up to 30 symbols or line segments can be drawn in one schematic using this program.

With this program, you use the pointer (graphic cursor) to select starting and ending points where a symbol is to be located. Then you select the symbol itself. You use specified User Definable Keys to move the pointer (see the programming tip in this issue title "Pointing To Areas On The Screen" for more information on using the User Definable Keys to move the pointer). This program also includes the facility to delete symbols or line segments, and to redraw the schematic after editing.



Schematic Drawing Program I

ABSTRACT NUMBER: 51/00-1402/0

Title: **Schematic Drawing Program II**
Memory Requirement: 16K
Peripherals: Hard Copy Unit Optional
Abstract:

This program operates the same as the Schematic Drawing Program I described above. This program, however, offers a choice of eight standard electronic symbols plus a line segment: resistor, capacitor, coil, diode, transistor, MOSFET, variable capacitor and earth (ground). Up to 50 symbols or line segments can be drawn in one schematic using this program.

(Abstracts continued on next page.)

● MISCELLANEOUS
INTERFACING

ABSTRACT NUMBER: 51/00-6101/0

Title: Hardware Modification of Hewlett Packard 59306A Relay Actuator for Use with the TEKTRONIX 4051

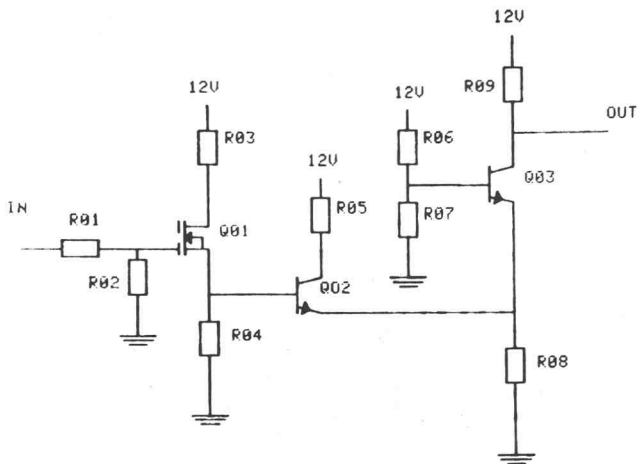
Author: Tyler Coplen

Memory Requirement: 8K

Peripherals: Hewlett Packard 59306A Relay Actuator

Abstract:

This program uses the 4051 as a controller to open and close each of six independent double-throw relays in the HP 59306A Relay Actuator. The HP 59306A is designed to operate on the HPIB (Hewlett Packard's implementation of the IEEE Standard 488-1975). With a minor modification of the HP 59306A hardware, it can be controlled with the 4051. The hardware modification consists of connecting REN (remote enable of the HP 59306A) to common. The only drawback of this modification is that the HP 59306A must be powered off and on to exit from the Local Lockout (LLO) mode.



Schematic Drawing Program II



TEKTRONIX®

TEKTRONIX, INC.
Information Display Group
Applications Library 60-369
P.O. Box 500
Beaverton, Oregon 97005

ADDRESS CORRECTION REQUESTED