SOFTWARE ANALYSIS OF CAPACITANCE-VOLTAGE MEASUREMENTS

Richard A. German
5th Year Microelectronic Engineering Student
Rochester Institute of Technology

ABSTRACT

A computer program called CVPLOT, used at RIT to aid in the analysis of metal-oxide-semiconductor (MOS) capacitors, was recoded from REGIS graphics into the VAX Graphical Kernel System (GKS) allowing the user to obtain hardcopy plots of high and low frequency capacitance voltage curves from an HP-Plotter, LNO3 Laser Printer, and LA100 Line Printer. The revised program also allows changes to be made in the values of the 'non-ideal' (a nonzero flatband shift) parameters while subsequently observing the shift in the curve as a result of these changes. Curves for three different values of either substrate doping, gate oxide thickness, or temperature may also be superimposed on the same graph.

INTRODUCTION

The capacitance-voltage (C-V) measurement is one of the most important techniques utilized in the analysis of the metal-oxide-semiconductor (MOS) structure. Information such as the threshold voltage, flatband voltage, oxide capacitance, oxide thickness, substrate doping, mobile ion concentration, distribution of surface states, and minority carrier lifetimes may be obtained to characterize both materials and the processing sequence.

A menu-driven in-house computer program called CVPLOT models capacitance voltage measurements for MOS capacitors based upon input entered from the terminal. Theoretical high and low frequency 'ideal' and 'non-ideal' (a nonzero flatband voltage shift) capacitance voltage curves can be displayed on the computer screen as well as superimposing experimental curves on the same graph. These theoretical curves are generated from the exact charge analysis theory for the MOS capacitor located in Reference 1. The program was limited in that the only method of obtaining a hardcopy output was through a screen dump to a graphics line printer.

The purpose of this project was to revise the existing CVPLOT program by expanding its capabilities in the following ways: to give the user the ability to direct the output to any of the following devices: computer screen, HP-Plotter, LA100 Line Printer, or the LNO3 Laser Printer; to permit defaults of commonly used RIT values for gate oxide thickness, gate area, temperature,

and substrate doping to be easily input during the execution of the program; and to allow for changes in Phims (nonzero metal/semiconductor workfunction difference), Qf (fixed oxide charge), and Qm (mobile oxide charge) during the running of the program while subsequently observing the shift in the 'non-ideal' curve on the screen as a result of these new values. Another goal was to superimpose 'ideal' C-V curves for three different values of either gate oxide thickness, substrate doping or temperature. The ultimate purpose was to improve the software tool available to the student for the analysis of capacitance voltage measurements.

PROGRAMMING

The graphic routines of the CVPLOT program were recoded from REGIS[2] graphics into the GKS[2,3] graphics format (see Appendix A for the programs). Figure 1 is a block diagram of the original CVPLOT program containing a description for each subroutine.

To implement the above changes, revisions were made to: the calling program, CVPLOT.FOR, and subroutines PLOTSUB.FOR, and CVSUB.FOR. In addition, two subroutines were added to the program, COPYSUB.FOR and HP_TRUNCATE.PAS. COPYSUB.FOR creates the output data files used for obtaining hardcopies of the results seen on the screen based upon the variables sent to it from CVPLOT.FOR. Problems developed in obtaining output on the HP-Plotter from the HP.DAT file created in COPYSUB.FOR. When this file was created, each line exceeded 80 characters resulting in arbitrary carriage returns and form feeds being injected into the file. To correct this, a subroutine called HP TRUNCATE.PAS was written in Pascal which takes the HP-Plotter file (HP.DAT) and reformats the strings to 80 characters per line so that they may be successfully plotted.

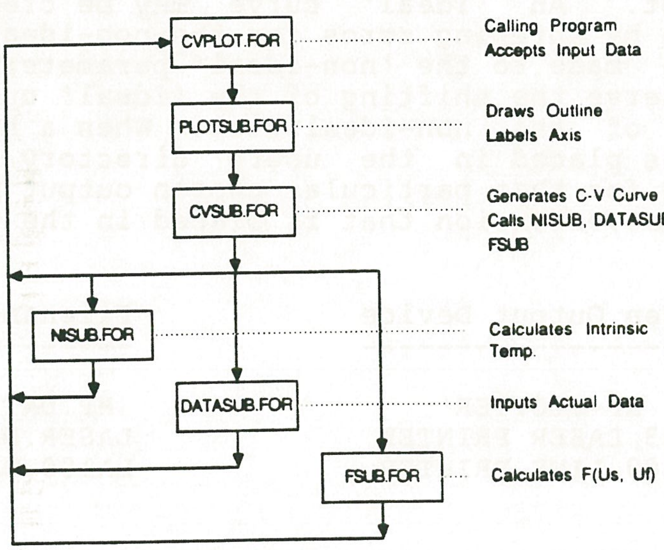

Figure 1: Block diagram for the original CVPLOT program.

One aspect of the recoding is that all of the output had to be transformed into graphics mode. This means that real numbers needed to be converted into character strings that could be handled by the graphics package. This was accomplished using a VAX Run-Time Library Routine called OTS$CNVOUT[4] which converted real numbers into character text in exponential notation. The data sheet for OTS$CNVOUT is included in Appendix B.

The revised CVPLOT program, which retains the original name, may be accessed from the following run command for VAX/VMS:

$RUN USER:[MICROLIB.TOOLS.CV]CVPLOT

where MICROLIB is the computer account in the microelectronic engineering department where the program resides. The user may desire to place the following command line in a LOGIN.COM file to eliminate having to repeatedly type the previous run command:

$CVPLOT:==RUN USER:[MICROLIB.TOOLS.CV]CVPLOT

By typing just CVPLOT at the dollar ($) prompt the program will be executed. Once in the program, the user chooses from the main menu the option to create 'ideal,' multivariable, or 'non-ideal' C-V curves. If the 'ideal' or multivariable route is chosen, then data is input in response to prompts at the computer terminal and the output is displayed on the screen. Next, the user has the option to create a hardcopy output file or return to the main menu.

If the 'non-ideal' path is chosen, values are again input and the graph is displayed. Next, the user has the option to change the 'non-ideal' parameters, i.e. Phims, Qf, or Qm. These changes may be executed as many times as desired, but if a hardcopy output is finally chosen then the last input values will be used to create the output plot. An 'ideal' curve may be created through the 'non-ideal' path by entering zeros for the non-idealities, and then changes may be made to the 'non-ideal' parameters. In this way, the user can observe the shifting of the 'ideal' curve in response to the effect of the non-idealities. When a hardcopy ouput is chosen, a file is placed in the users directory containing the appropriate data for that particular chosen output device. Table 1 gives the filename.extension that is placed in the users directory,

Chosen Output Device	Filename
HP-PLOTTER	HP.DAT
LN03 LASER PRINTER	LASER.DAT
LA100 LINE PRINTER	LA100.DAT

Table 1: Listing of filename.ext placed in users directory from the CVPLOT program.

RESULTS

Figure 2 outlines the block diagram of the revised CVPLOT program with the two previously mentioned subroutines. Figure 3 is an example of a normalized 'ideal' curve output to the LN03 Laser Printer. Figure 4 illustrates an 'ideal' graph output on the HP-Plotter for three different values of substrate doping. Figure 5 demonstrates the combination of a 'non-ideal' shifted curve with experimental data (dashed curve) plotted on the LA100 Line Printer. Notice that when an 'ideal' low frequency curve is plotted, the threshold voltage is also extracted for either the high or low frequency experimental data curve. This plot only uses the experimental data for demonstration purposes and no relationship between the two curves is intended.

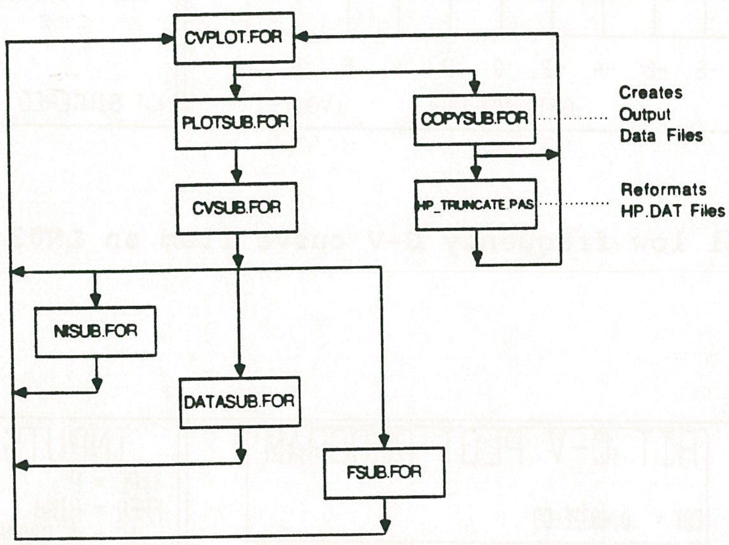

Figure 2: Block diagram for the revised CVPLOT program.

CONCLUSION

The revised CVPLOT Program is fully operational at this time. Hardcopy output can be obtained from the HP-Plotter, LN03 Laser Printer and the LA100 Line Printer. Suggestions for further work include modifying the COPYSUB.FOR routine to allow plotting of 'ideal' and 'non-ideal' curves on the same graph. This is only available now through the screen dump. The effects of sintering on the curves along with calculations for the interface trap density are future enhancements that would be of great benefit.

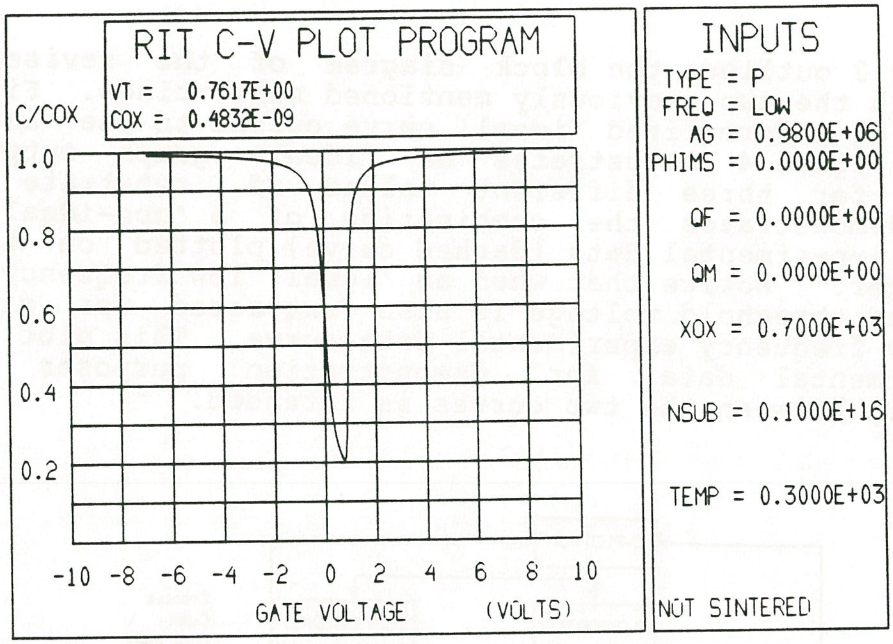

Figure 3: Ideal low frequency C-V curve from an LN03 Laser Printer.

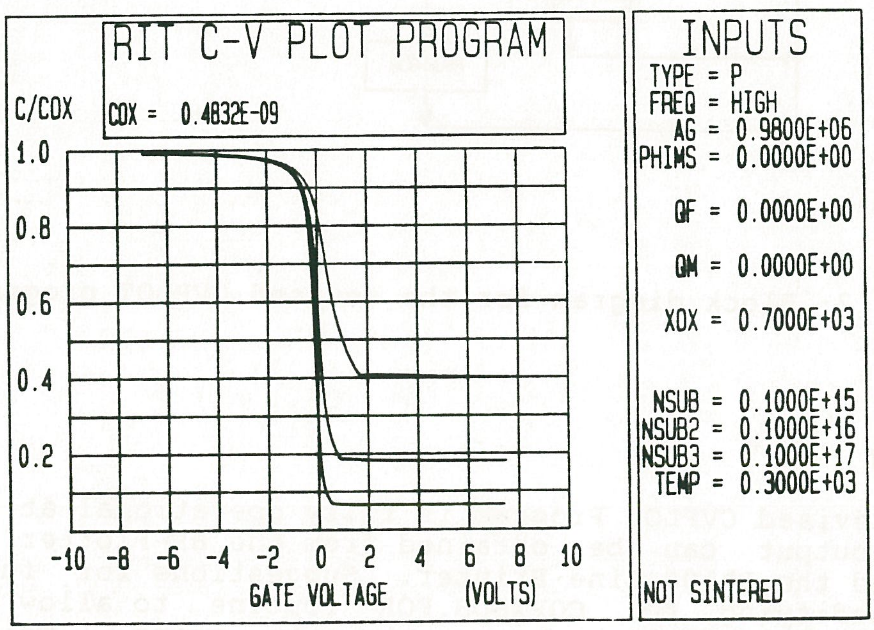

Figure 4: Ideal high frequency curves for three values of substrate doping output from an HP-Plotter.

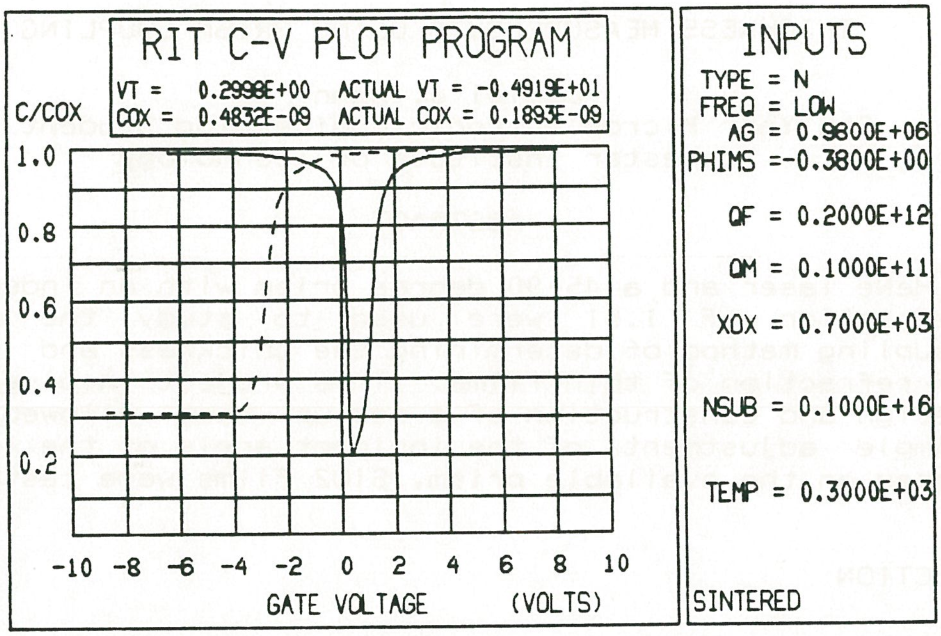

Figure 5: Non-ideal low frequency curve with experimental data curve output from an LA100 Line Printer.

ACKNOWLEDGMENTS

The author would like to acknowledge the help of Steve Wilkins of the RIT User Computer Center for his assistance with GKS graphics, Mike Young, computer lab assistant with the UCC for his HP-TRUNCATE.PAS subroutine, Robert Pearson for help on deciphering his original CVPLOT program, and Mike Jackson for continued support.

REFERENCES

[1] Robert F. Pierret, Modular Series on Solid State Devices, Volume IV, (Addison-Wesley Publishing Company, Reading, MA, 1983), pp. 50.

[2] Copyright Digital Equipment Corporation

[3] VAX/VMS Graphical Kernel System (GKS) Manuals

[4] VAX/VMS Run-Time Libray Manuals