# Series 682XXB/683XXB Synthesized Signal Generator Instrument Driver for LabWindows

# User's Guide

Version 1.00

Locate the Serial Number decal sheet packaged with the manual, and paste the driver software serial number here.

**Serial Number**

You will be asked for this number when you call ANRITSU Customer Service for support.

***NOTICE***

ANRITSU Company was formerly known as ANRITSU Company.

Anritsu

490 JARVIS DRIVE
MORGAN HILL, CA 95037-2809

P/N: 10570-00015
REVISION: B
PRINTED: FEBRUARY 1998
COPYRIGHT 1994 ANRITSU COMPANY

### *Limited Warranty*

The media on which you receive ANRITSU Company software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. ANRITSU Company will, at its option repair or replace software media that do not execute programming instructions if ANRITSU Company receives notice of such defects during the warranty period. ANRITSU Company does not warrant that the operation of the software shall be uninterrupted or error free.

EXCEPT AS SPECIFIED HEREIN, ANRITSU COMPANY MAKE NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF ANRITSU COMPANY SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. ANRITSU COMPANY WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

The aforestated limitation of the liability of ANRITSU Company will apply regardless of the form or action, whether in contract or tort, including negligence. Any action against ANRITSU Company must be brought within one year after the cause of action accrues. ANRITSU Company shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's abuse, misuse, or negligence acts; and power failure or surges, fire, flood, accident, actions or third parties, or other events outside reasonable control.

# *Copyright*

Under the copyright laws, this book may not be copied, photocopied, reproduced, translated — in whole or in part — without the written consent of ANRITSU Company.

Copyright 1994 ANRITSU Company

# *Trademarks*

LabWindows is a registered trademark of National Instruments Corporation.

IBM is a registered trademark of International Business Machines Corporation. Personal System/2, IBM PC, PC AT, PC/XT, PC DOS, IBM CGA, IBM EGA, IBM VGA, and Micro Channel are trademarks of International Business Machines Corporation.

Microsoft, Microsoft QuickBASIC, Microsoft BASIC, Microsoft Visual BASIC, and Microsoft C are trademarks of Microsoft Corporation.

# *Preface*

The 682XXB/683XXB LabWindows Instrument Driver User's Guide provides a tutorial and both general and detailed descriptions of the various functional panels displayed in the LabWindows environment. The user should be familiar with measurements using the applicable ANRITSU instrument and with MS- or PC-DOS conventions. A knowledge of LabWindows, while helpful, is not essential. The ANRITSU Instrument Drivers software can be used to create executable stand-alone application programs.

# *Manual Organization*

The manual is divided into three sections:

Section 1, General, provides general information about the LabWindows program.

Section 2, Using the 682XXB/683XXB Driver with LabWindows, provides description and a tutorial for using the driver within the LabWindows environment.

Section 3, Driver References, provides detailed descriptions of the function panels and instrument controls. It also provides sample syntax and a listing of variable-type used in the program.

# Table of Contents


## *Section 1 — Introduction to 682XXB/683XXB Instrument Driver for LabWindows*

**General . . . . . 1-3**

**Requirements . . . . . 1-3**

**Installing Instrument Drivers . . . . . 1-4**

**Overview For LabWindows Users . . . . . 1-4**

**Overview for Non-LabWindows Users . . 1-5**

## *Section 2 — Using the 682XXB/683XXB Instrument Driver with LabWindows*

**Introduction . . . . . 2-3**

**General . . . . . 2-3**

**Loading 682XXB/683XXB Driver as Instrument Module . . . . . 2-4**

**Loading 682XXB/683XXB Driver At Start-up . . . . . 2-6**

**Debug Utility . . . . . 2-8**

**Tutorial . . . . . 2-9**

**Creating a Compiled Program . . . . . 2-17**

## *Section 3 — Driver References*


**Introduction . . . . . 3-3**

**Function Panel Descriptions . . . . . 3-3**

**(Function Panels and Syntax)**

**Close (close) . . . . . 3-6**
**Alternate Sweep (FC.ALS) . . . . . 3-8**
**CW Sweep (FC.CW) . . . . . 3-10**
**Delta-F Sweep (FC.DELTA) . . . . . 3-12**
**Special Sweep (FC.SPEC) . . . . . 3-14**
**Start-Stop Sweep (FC.STST) . . . . . 3-16**
**Initialize (init) . . . . . 3-18**
**Set Frequency Definitions (INIT.FREQ) . . . . . 3-20**
**Set Power Definitions (INIT.PWR) . . . . . 3-22**
**Output Power Level (PC.LEVEL) . . 3-24**
**Output Power (PC.PWR) . . . . . 3-28**
**User Power Calibrate (PC.PWR-CAL) . . . . . 3-30**
**Output Power Sweep (PC.PWRS) . . 3-32**
**Amplitude Modulation (SETAM) . . 3-34**
**Frequency Setup (SETCW) . . . . . 3-38**
**Frequency Modulation (SETFM) . . 3-40**
**Marker Selection (SETMK) . . . . . 3-44**
**Pulse Modulation (SETPM) . . . . . 3-46**
**Recall Setup (SR.RECALL) . . . . . 3-50**
**Save Setup (SR.SAVE . . . . . 3-52**
**Utilities (UT) . . . . . 3-54**

# Section 1

# Introduction to the 682XXB/683XXB Instrument Driver for LabWindows

# Section 1
# Introduction to the 682XXB/683XXB Instrument Driver for LabWindows

## *General*

ANRITSU Instrument Driver software provides an easy-to-use tool for developing application programs for applicable microwave systems via the General Purpose Interface Bus (IEEE-488 Bus).

This software contains modules that automatically configure an applicable ANRITSU instrument for use on the bus, along with high-level instrument control commands that save you the time required to learn and program the GPIB commands of the instrument. The software automatically checks for proper bus functioning. If a command is sent to a bus instrument and no error is reported, the bus can be assumed to be working correctly.

## *Requirements*

The ANRITSU Instrument Driver software is written specifically for the model 682XXB or 683XXB.

The ANRITSU Instrument Driver requires an IBM PC AT, PS/2, or compatible computer running MS- or PC-DOS, Version 3.0 or later.

The software is delivered on 5-1/4 inch 1.2 Mb Floppy disks and 3-1/2 inch 1.4 Mb floppy disks.

At least 2 MB of memory is required to run the LabWindows program — 4 Mb is recommended.

The ANRITSU Instrument Driver software requires National Instruments LabWindows version 2.2 or later.

For Microsoft QuickBASIC*, Professional BASIC, Visual BASIC for DOS, C, Quick C, and Borland C++ and Turbo C++ users, you can use the 682XXB/683XXB Instrument Driver software to produce compatible instrument-control-program code.

* ANRITSU strongly recommends that QuickBASIC not be used. Instead, use Microsoft Professional BASIC 7.1, or later, or Visual BASIC for DOS.

# *Installing Instrument Drivers*

This section provides instructions for installing the ANRITSU Instrument Driver. Proceed as follows:

- ☐ Insert the ANRITSU driver diskette in your A: or B: drive, as appropriate.
- ☐ Change to the LabWindows, Instruments directory (*drive*\LW\INSTR), and type the following DOS command: COPY A: (B:) *.*. This copies the following four files to the target subdirectory: W683B.LBW, W683B.LWI, W683B.FP, W683B.DOC (All four of these files MUST reside within the same subdirectory.)
- ☐ Once the copying is completed, return the driver diskette to a safe storage location.

# *Overview For LabWindows Users*

LabWindows is a software development system for BASIC, C, and C++ programs (see page 1-3 for listing of supported languages). It contains an interactive environment for developing programs with drivers and libraries (functions) for creating data acquisition and instrument control applications. LabWindows contains a comprehensive set of software tools for data analysis, data presentation, and high level instrument control.

The interactive program is an environment for editing and debugging BASIC and C (C++) programs. In the LabWindows environment, you can use the functions in the instrument drivers or libraries to write your program. In addition each function has an interface called a function panel that lets you interactively execute the function or generate code for calling the function.

The interactive program uses extended memory. Programs executed in the interactive program can use up to 16 megabytes of memory, depending on your computer configuration. Programs that run in the interactive program, however, must adhere to the LabWindows subsets for BASIC, C and C++.

Programs developed with the drivers and library functions can be run within the interactive program, or they can be compiled and linked into a stand-alone applicaton (*.EXE) or run-time application (*.RTM) file. To help you create a stand-alone program, LabWindows incorporates utilities that automate the compile and link processes.

The real power of LabWindows lies in the libraries. They have functions for developing all phases of your data acquisition and instrument control system. For controlling the 682XXB/683XXB, Lab Windows has the Instrument Drivers Library. The programs that call this library can be developed with the interactive program. This program has tools that make program development quicker and easier.

LabWindows gives you the capability to execute instrument drivers with the aid of panels and thereby create programs easily. The panels contain items that can be selected to build and execute a driver. The drivers are separately declared in the Instrument Drivers Library.

Two advantages of using LabWindows are:

- ☐ When writing an application program you do not have to remember all of the parameters that belong to the driver.
- ☐ Error reporting is shown automatically in the panels.

# *Overview for Non-LabWindows Users*

Programmers who do not use LabWindows will also benefit from the ANRITSU Instrument Driver software:

- ☐ You will not have to know all of the GPIB codes needed to program applications for the 682XXB or 683XXB Synthesized Signal Generator. The driver software effectively manages low-level GPIB I/O operations and native instrument control.
- ☐ You will see greater program reliability because of the driver's extensive error-checking routines.
- ☐ You will see reductions in the time required to develop, test, and debug applications.

# Section 2

# Using the 682XXB/683XXB Instrument Driver with LabWindows

# Section 2
# Using the 682XXB/683XXB Instrument Driver with LabWindows

## *Introduction*

This section provides an introduction to the LabWindows environment and a tutorial describing the use of the 682XXB/683XXB driver within LabWindows. This section assumes that you have read Part 1 of the National Instruments *Getting Started with LabWindows* manuals and are generally familiar with the LabWindows screen and principles of navigation within the environment.

## *General*

The following procedure describes how to access LabWindows and load files.

- ☐ Move to the directory containing the LabWindows executable (*.EXE) files. (This directory is usually named \LW.)
- ☐ Type LW.
  This places you in the PROGRAM window of the LabWindows environment (below).

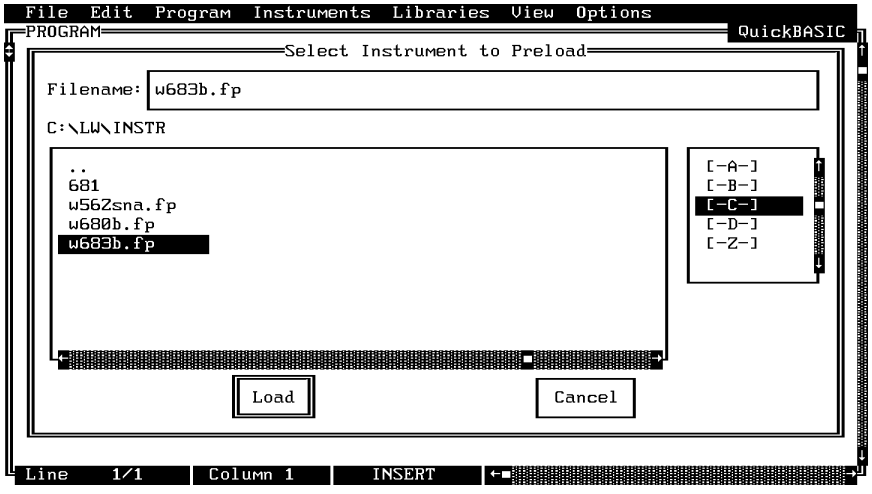
# *Loading 682XXB/683XXB Driver As Instrument Module*

The following procedure describes how to load the 682XXB/683XXB Driver as an instrument module.

☐ Select **Instruments,** on the top menu bar (below), to display the pull-down menu.

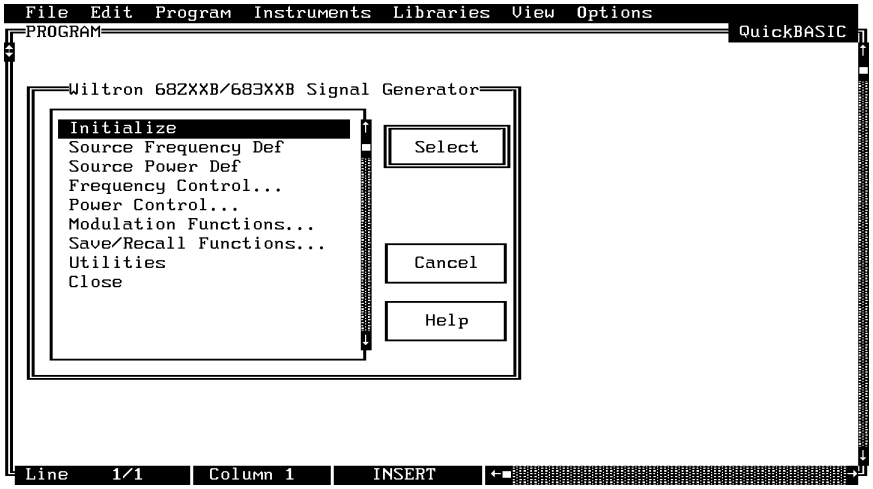
Select **Load,** then change to the \LW\INSTR subdirectory (below).

☐ Move the cursor to **w683b.fp** and select **Load.**



☐ Move the cursor to **Instruments** to display the pull-down menu. The "ANRITSU 682XXB/683XXB Signal Generator" title displays showing that the instrument module is now loaded.

☐ Select **ANRITSU 682XXB/683XXB Signal Generator,** and observe that the main panel appears (below).

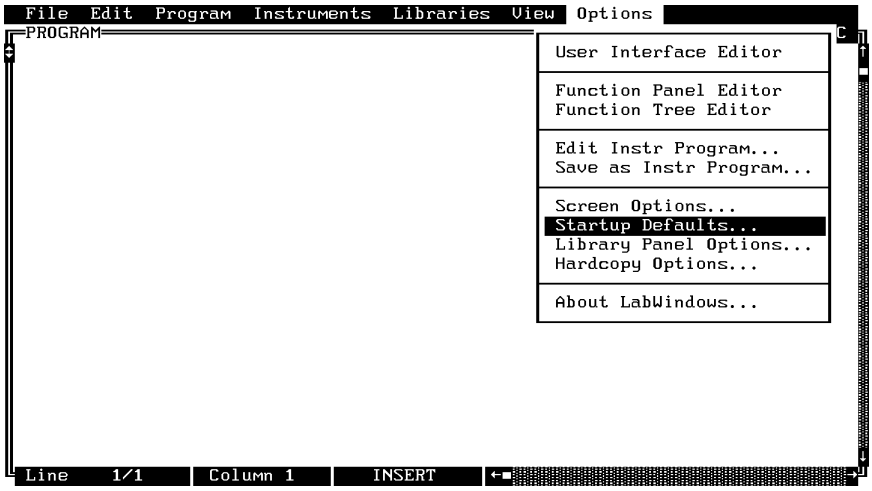



☐ You are now ready to proceed with developing control code using the 682XXB/683XXB instrument driver.

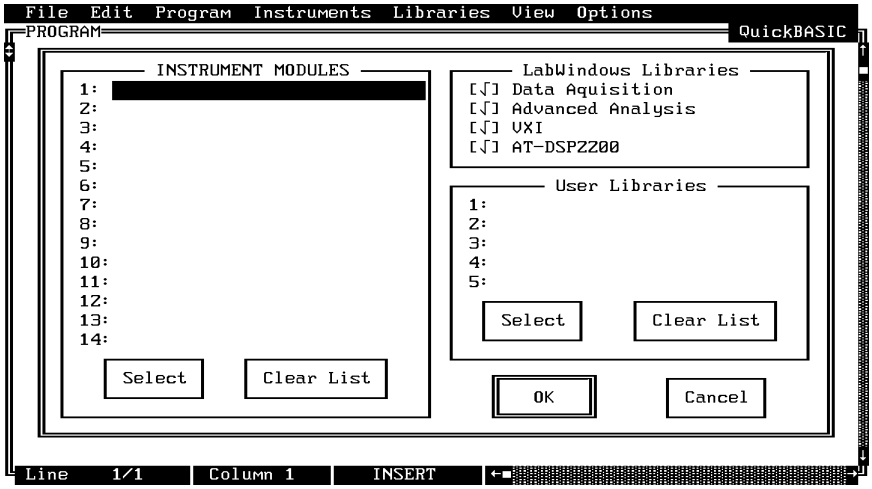
# *Loading 682XXB/683XXB Driver At Start-up*

The 682XXB/683XXB driver can be automatically loaded each time LabWindows is started. The procedure for making this happen is given below.

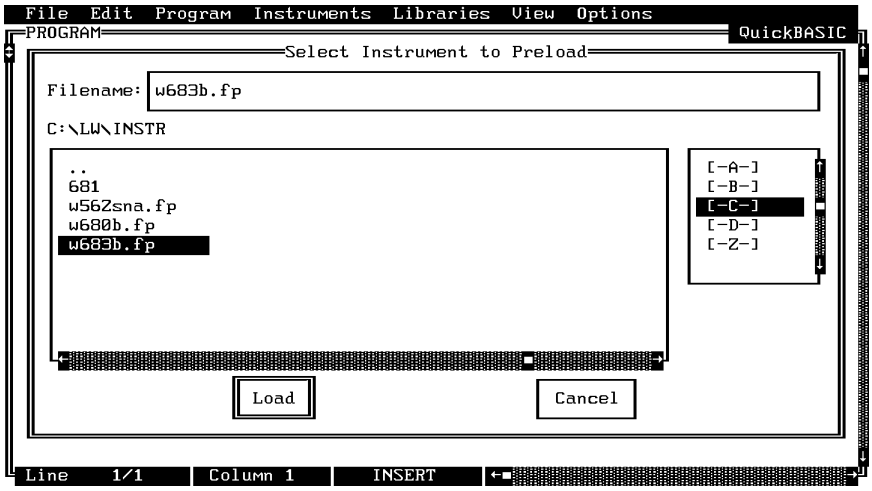
- ☐ Place yourself into the LabWindows PROGRAM window as was described on page 2-3.
- ☐ Select **Options**, on top menu bar, to display the pull-down menu.
- ☐ Select **Startup Defaults.**




- ☐ Choose **Select** to locate the 683XXB driver (w683b.fp); alternatively, type in the path and file name.

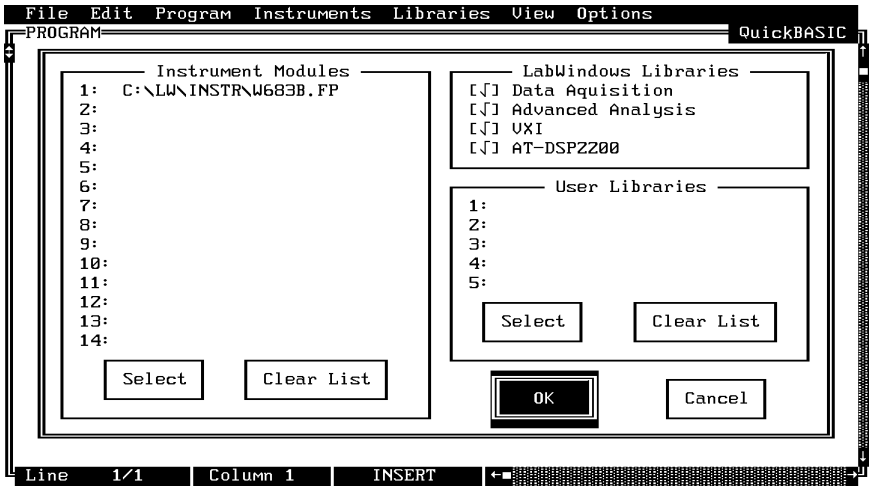


☐ Choose the **\LW\INSTR** subdirectory from the file list.



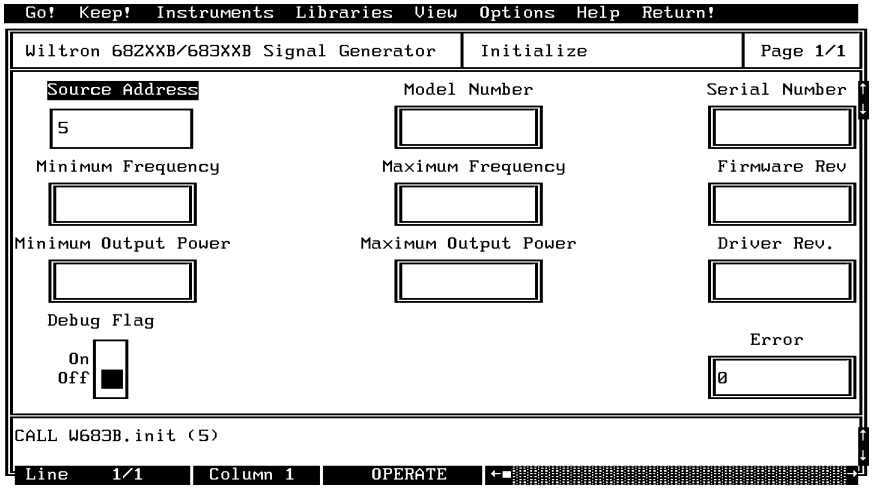

☐ Choose **w683b.fp** from the displayed file list, and click on **Load.**

☐ The appropriate path will now be displayed in the INSTRUMENT MODULES window. Select **OK** to exit. The W683 driver will now be loaded automatically each time LabWindows is entered.



# *Debug Utility*

The 682XXB/683XXB LabWindows driver incorporates a unique Debug utility, which should be used for program development only. It should be disabled when compiling stand alone applications.




Error 305 is returned when the instrument has generated an unexpected SRQ. To determine the cause of the SRQ set the **Debug** switch to **On** (Debug%=1), the driver will respond with a specific 3XX error code (Table 2-1).

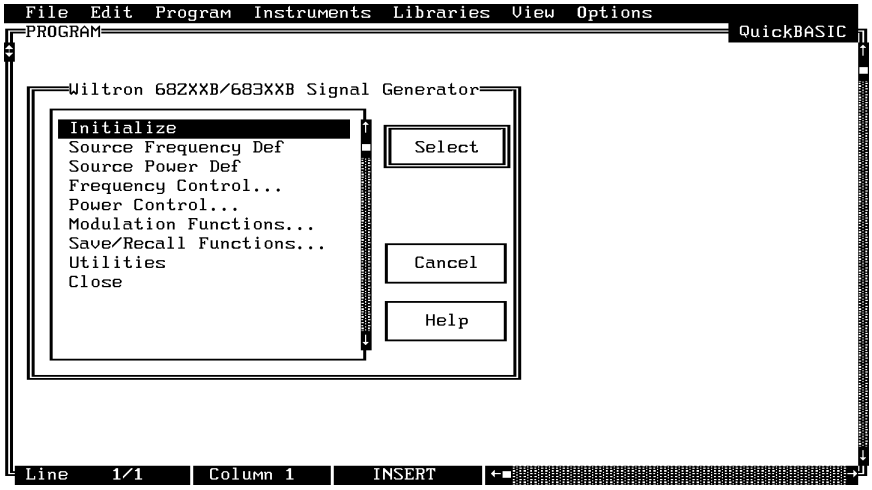
**Table 2-1.** Error Codes

| Error Code | Description |
|---|---|
| 301 | Parameter Range Error |
| 302 | File Not Found |
| 310 | Disk Failure |
| 311 | Self Test Failed |
| 312 | Hardware Error |

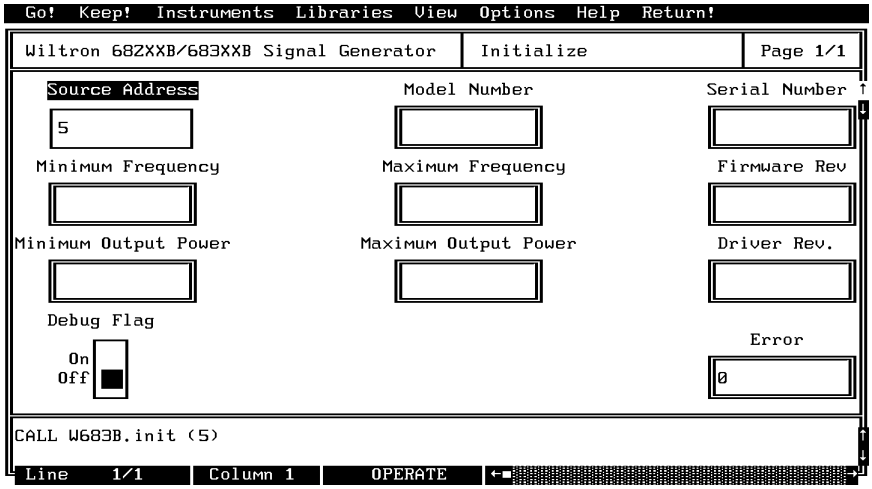
# *Tutorial*

This tutorial takes you through the development of a simple program. This program initializes the 682XXB/683XXB Signal Generator, assigns sweep frequencies and modulation parameters, and closes the driver. When you have finished stepping through the tutorial, you will have a BASIC program that can be compiled as a DOS executable (*.EXE) file. A Microsoft C program could be produced in the exact same manner by switching the native language to C (under the **Program** menu).

☐ Move cursor to **Instruments** in the top menu bar, and select **ANRITSU 682XXB/683XXB Signal Generator.** This will display the pull-down menu shown below.



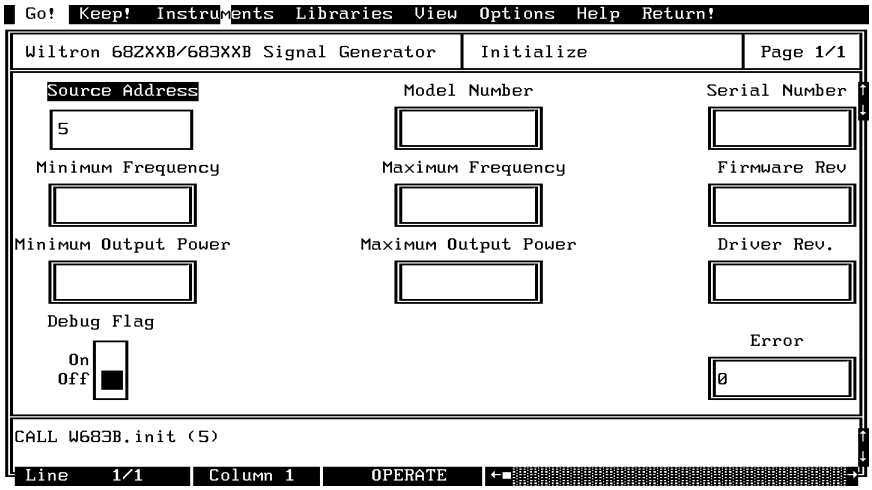

☐ Choose **Select** to display the Initialize panel (below).



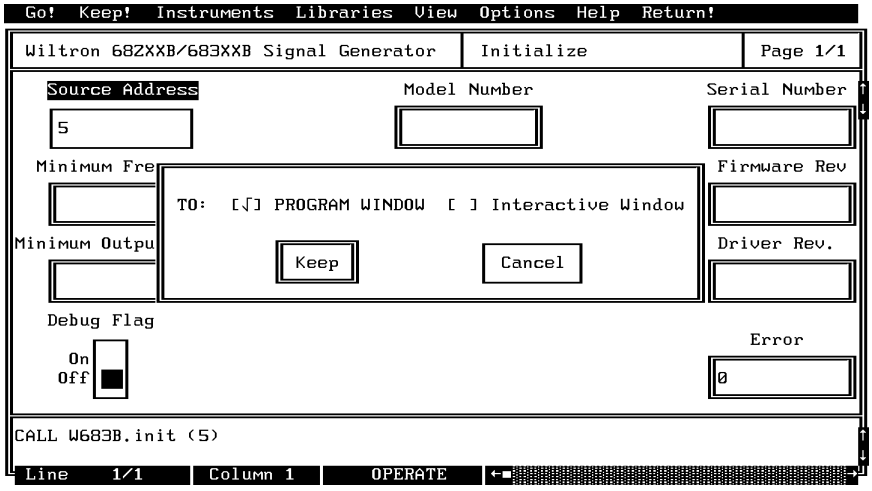

☐ With the **Source Address** highlighted as shown, enter the 682XXB/683XXB GPIB address (a number between 1 and 30; 5 is the default).

☐ Select **Go!** (below) in the top menu bar. If a 682XXB or 683XXB is connected and set to GPIB address 5, it will respond by resetting itself and returning an identification string that will fill the **Model Number, Firmware Rev, Minimum Frequency, Maximum Frequency, Serial Number, Minimum Output Power,** and **Maximum Output Power** fields.

☐ Leave Debug Flag set to Off. This switch was discussed on page 2-8.

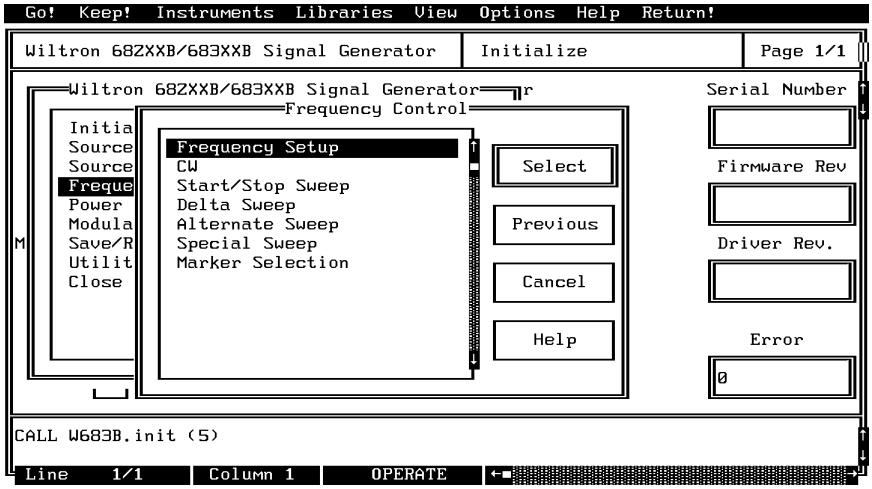



☐ Move the cursor to **Keep!**, in the top menu bar, then choose **Keep** from the next window (below) to select the default option. This transfers the code shown at the bottom of the panel to the PROGRAM window of the LabWindows environment.




☐ Select **Instruments,** in the top menu bar, then **ANRITSU 682XXB/683XXB Signal Generator** to return to the 682XXB/683XXB Driver main panel.

☐ Select **Frequency Control,** from the main menu, then **Frequency Setup** when the Frequency Control menu appears (below).

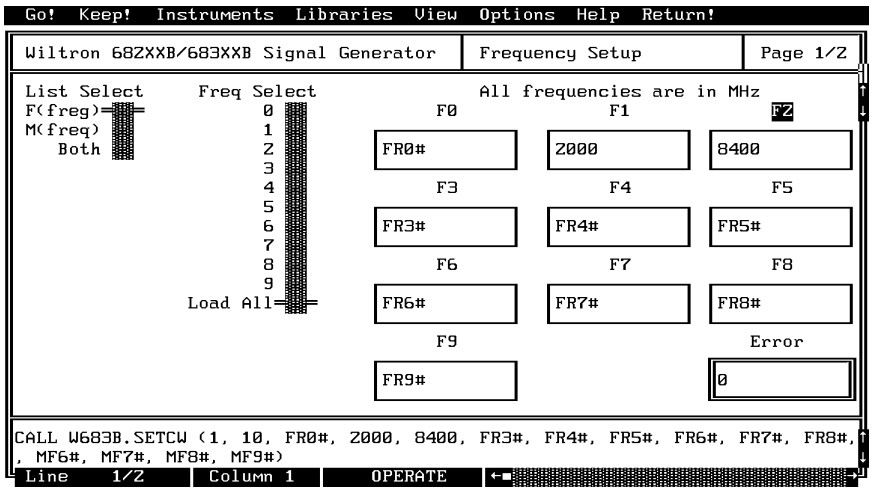



☐ Select **F(freq)** (below), from the **List Select** control.

☐ Select **Load All,** from the **Freq Select** control.

☐ **Select F1,** and enter **2000** from the keyboard.

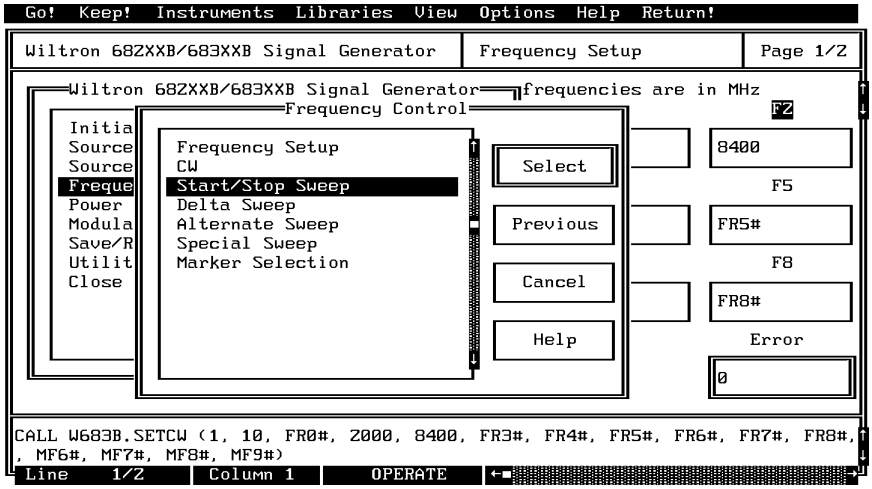
☐ With **F2** highlighted, enter **8400** from the keyboard.



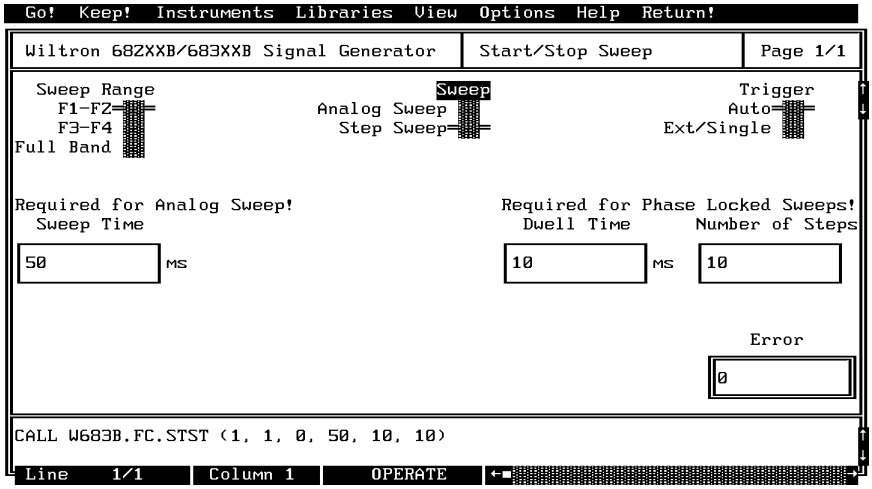

☐ Select **Keep!**, in the top menu bar; then choose **Keep** again when the next menu appears.

☐ Select **Instruments,** in the top menu bar, then **ANRITSU 682XXB/683XXB Signal Generator** to return to the 682XXB/683XXB Driver main panel.

☐ Select **Frequency Control** and **Start/Stop Sweep** from the next menus to appear (below).



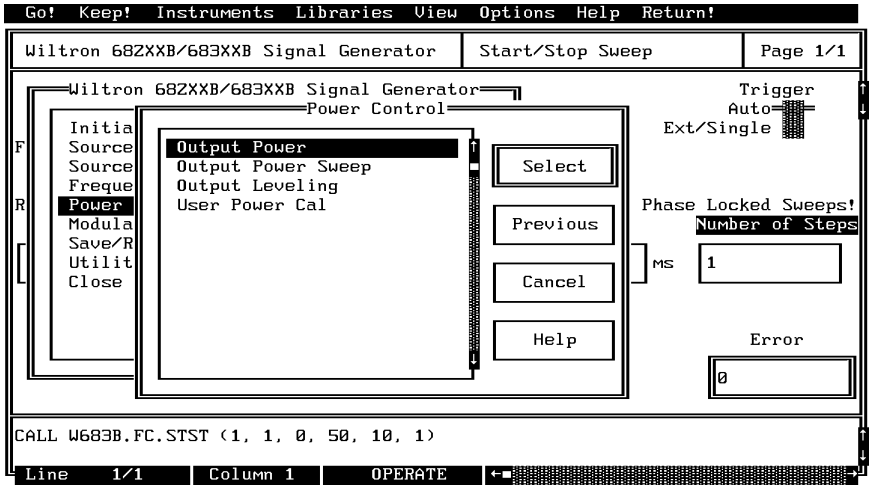

☐ Select **F1–F2** sweep range (below), **Step Sweep, Auto** trigger, and the default settings of **50** ms sweep time, **10** ms dwell time, and **10** steps.




☐ Select **Keep!,** in the top menu bar; then choose **Keep** again to add the code to the program window.

☐ Select **Instruments,** in the top menu bar, then **ANRITSU 682XXB/683XXB Signal Generator** to return to the main driver panel.

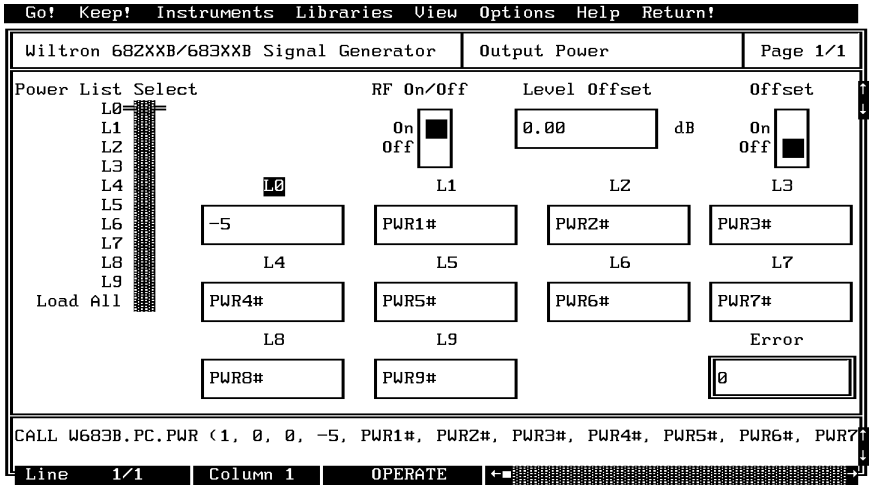
☐ Select **Power Control** then **Output Power** (below) to display the output power panel (next menu).




☐ Select **L0** on the **Power List Select** control (below)**.**

☐ Select **RF On/Off** and **Offset** to be **Off** (defaults).

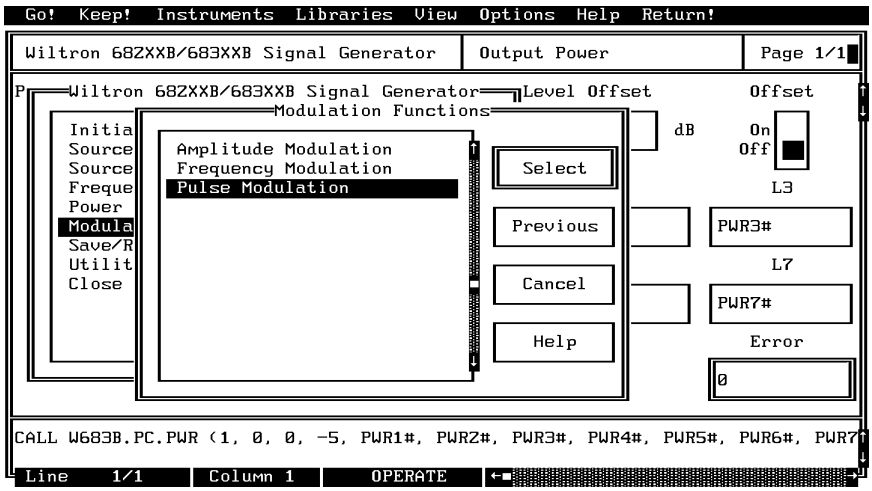
☐ Enter **–5** from the keyboard in the **L0** field.




☐ Select **Keep!,** in the top menu bar; then choose **Keep** again to add the code to the program window.

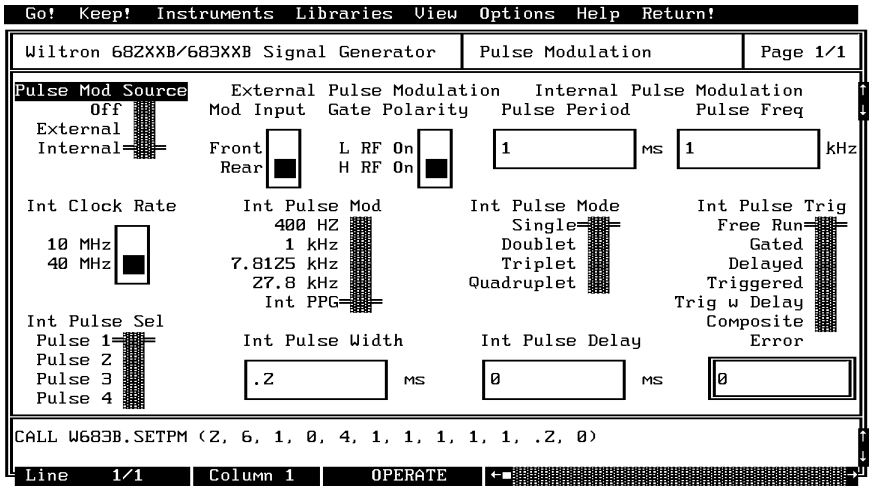
☐ Select **Instruments,** in the top menu bar, then **ANRITSU 682XXB/683XXB Signal Generator** to return to the main driver panel.

☐ Select **Modulation Functions** from the first menu, then **Pulse Modulation** from the second (below) to display the pulse modulation menu (below).




☐ Position **Pulse Mod Source** to **Internal** (below).

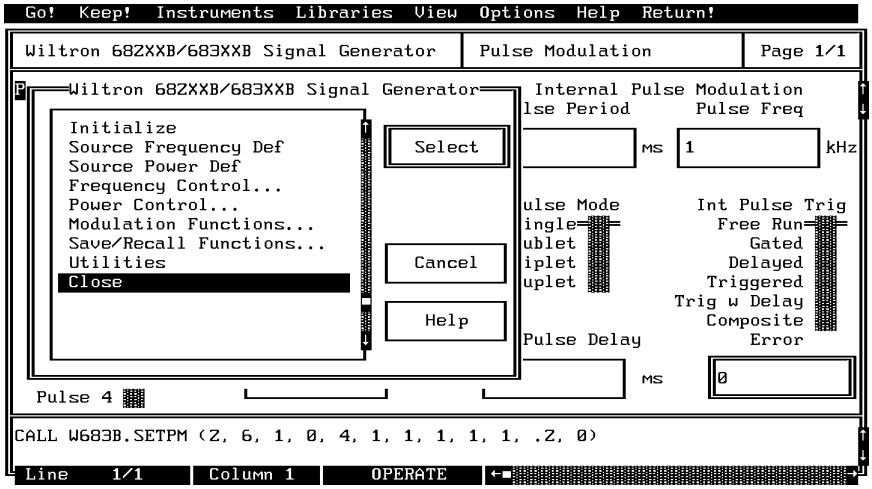
☐ Select **Rear, H RF On, 1 ms**, and **1 kHz** from the available menu options. Also, keep the default settings of **40 MHz, Int PPG, Single, Free Run, Pulse 1, .2,** and **0**, as shown below.



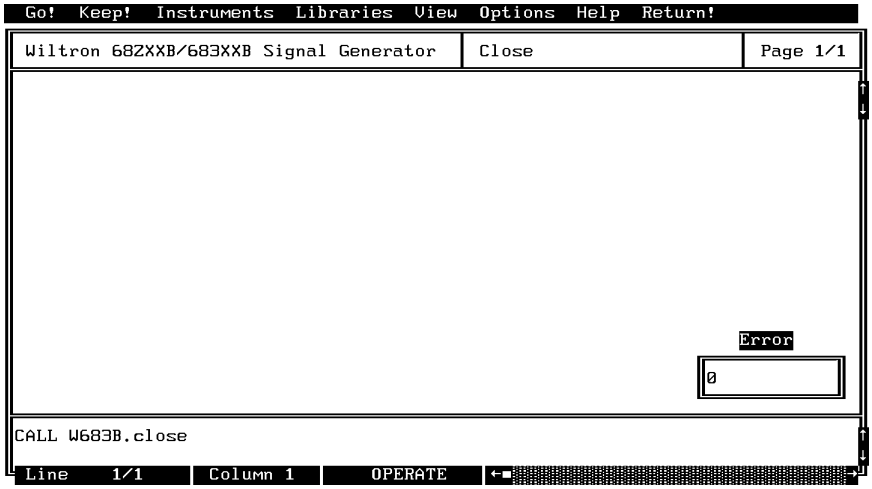

☐ Select **Keep!,** in the top menu bar; then choose **Keep** again to add the code to the program window.

☐ Select **Instruments,** in the top menu bar, then **ANRITSU 682XXB/683XXB Signal Generator** to return to the main driver panel.

☐ Move cursor to **Close** (below) and choose **Select** to display the Close panel

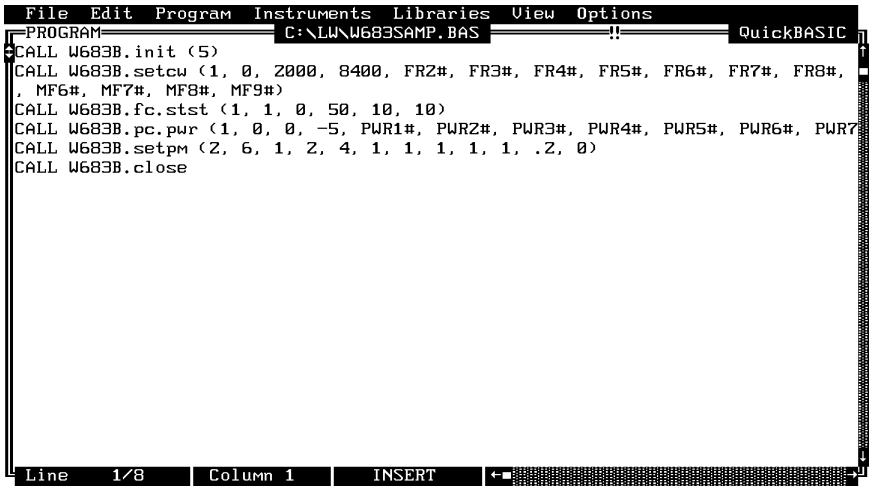



☐ There is nothing you need to do with this panel, except to select **Keep!,** in the top menu bar; then choose **Keep** again to add the code to the program window.

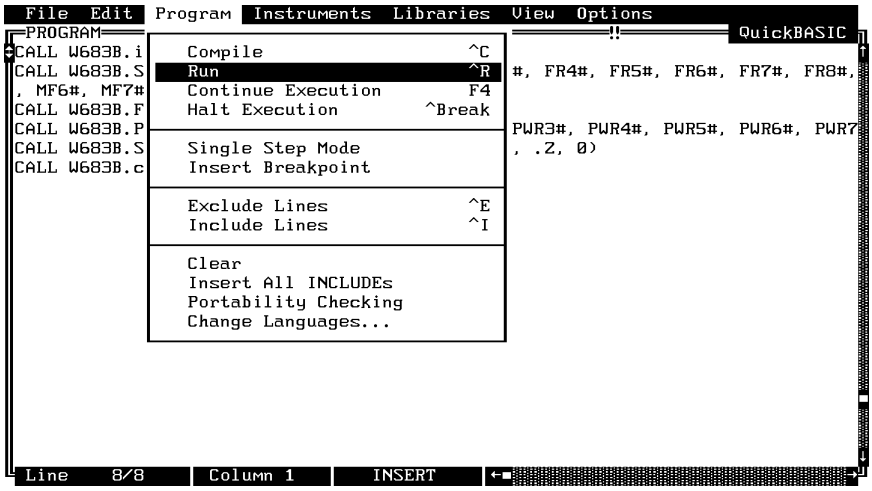



☐ Select **Return!** on the menu panel to return to the PROGRAM area.

☐ The program that you have developed now appears (below).




☐ To test the program, move the cursor to **Program,** in the top menu bar and select **Run** (below) from the pull-down menu.




☐ If there are no program errors triggering a syntax error message, you will see the applicable 682XXB or 683XXB Signal Generator respond to the program code.

☐ After you are satisfied that the program runs correctly, you can use the **Create.EXE** option, on the File menu, to create a stand-alone DOS executable (*.EXE) file. Alternatively, you can use the **Create RTM** option, on the file menu, to create a run-time version (*.RTM) file. That operation is described in the next heading.

☐ To continue with this tutorial, save this program as *\LW\PROGRAMS\W683SAMP.BAS*

# *Creating a Compiled Program*

Most programs developed with LabWindows can be compiled with the Microsoft C or BASIC compiler. Some applications, however, become too large to compile and run in the 640 KB DOS memory and must be run within LabWindows or the LabWindows Run-Time System. The LabWindows Run-Time System includes a DOS extender so programs can access up to 16 MB of memory during execution. Programs executed in the run-time system can make calls to any of the LabWindows libraries and instrument drivers. Programs distributed with the run-time system are in a binary format, so the programs cannot be edited.

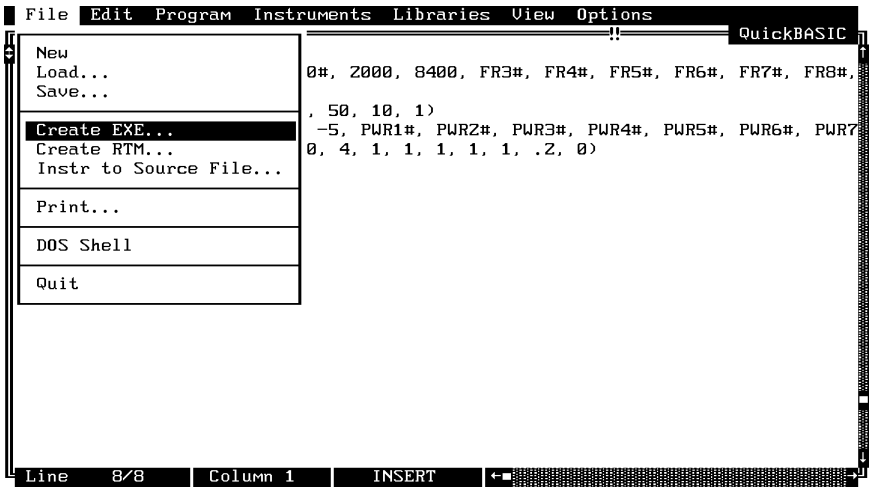
A stand-alone application (*.EXE or *.RTM) that incorporates the 682XXB/683XXB Driver may be created using the Microsoft C or BASIC compilers or LabWindows Run-Time System. To avoid OUT OF MEMORY errors when using the Microsoft QuickBASIC compiler, you must first optimize the 682XXB/683XXB Driver memory usage with the LabWindows FUNNEL.EXE utility (See the *LabWindows User's Manual* for instructions).

The LWMAKE option on the file menu can be used to create an executable file in either BASIC or C. The following provides a step-by-step tutorial for creating a *.EXE file using the BASIC compiler. This tutorial assumes that you have read and are familiar with the LWMAKE utility description in the *LabWindows User's Manual.*

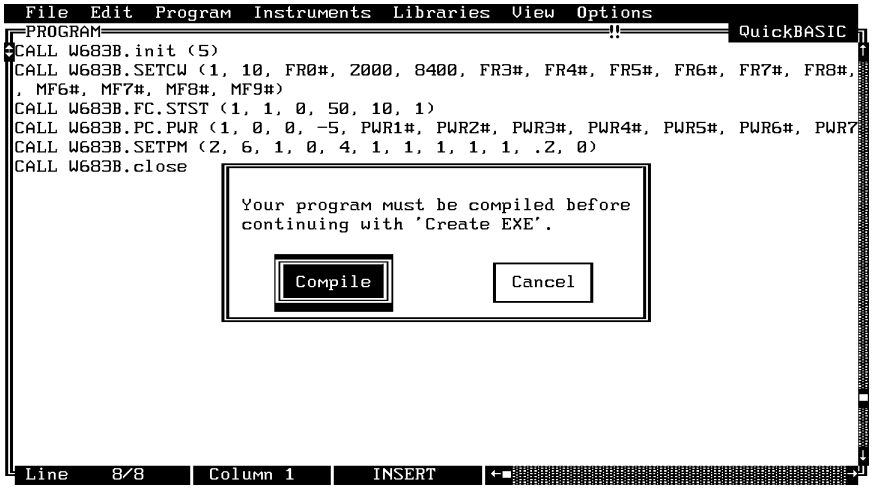
We will start with the program that you completed in the preceeding tutorial. If you did not complete the tutorial, you can type the program listing shown on the preceeding page, and save it as *drive*\LW\PROGRAMS\W683SAMP.BAS.

- ☐ Move the cursor to **Instruments,** in the top menu bar, and ensure that the **ANRITSU 682XXB/683XXB Signal Generator** driver is loaded. If it is not loaded, refer to pages 2-4 and 2-5 for instructions.

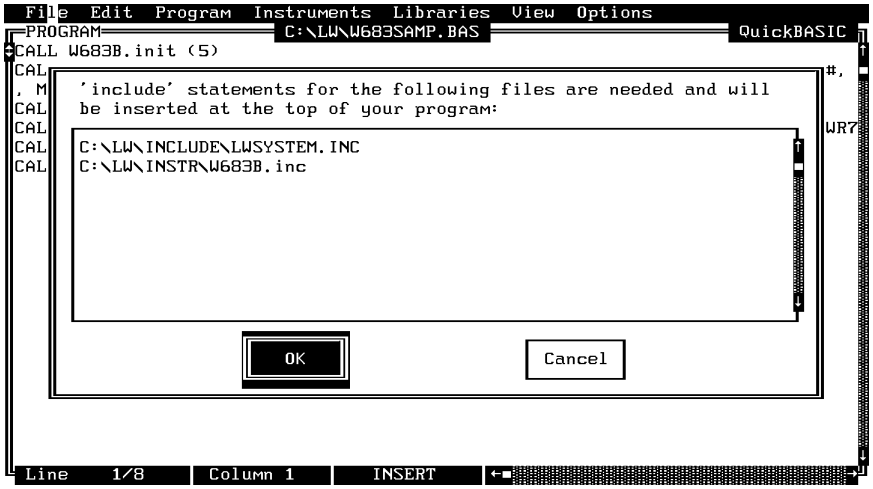
- ☐ Move the cursor to **File,** in the top menu bar, and select **Create EXE** from the pull-down menu (below).



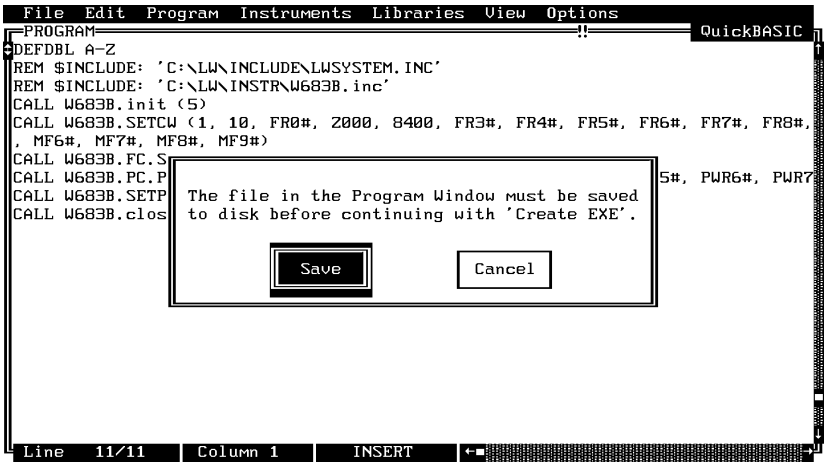
☐ Choose **Compile** (or **Check Syntax**, if you have run the program before), when the next prompt appears.



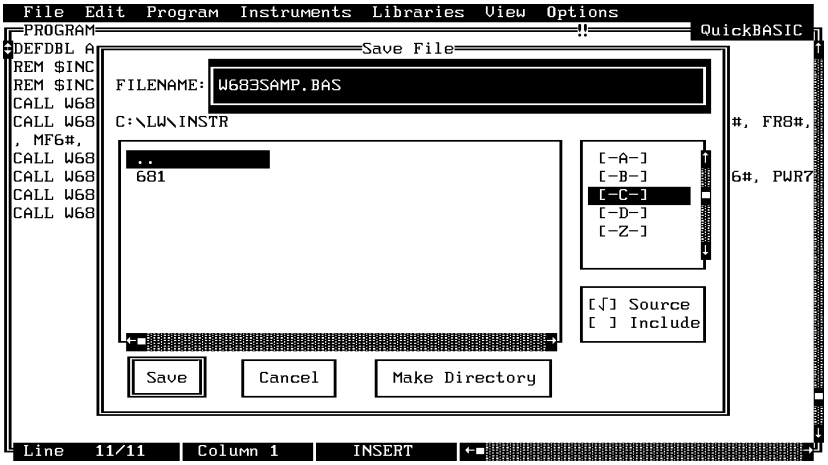

☐ Answer **OK** to the next prompt. This will include calls to the two required files shown in the file list. These files contain code needed to run your application.



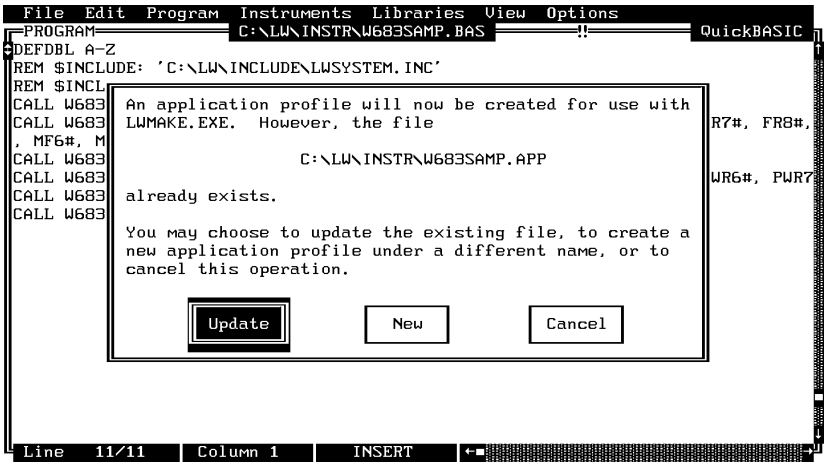
☐ Choose **Save** for the next prompt, below.



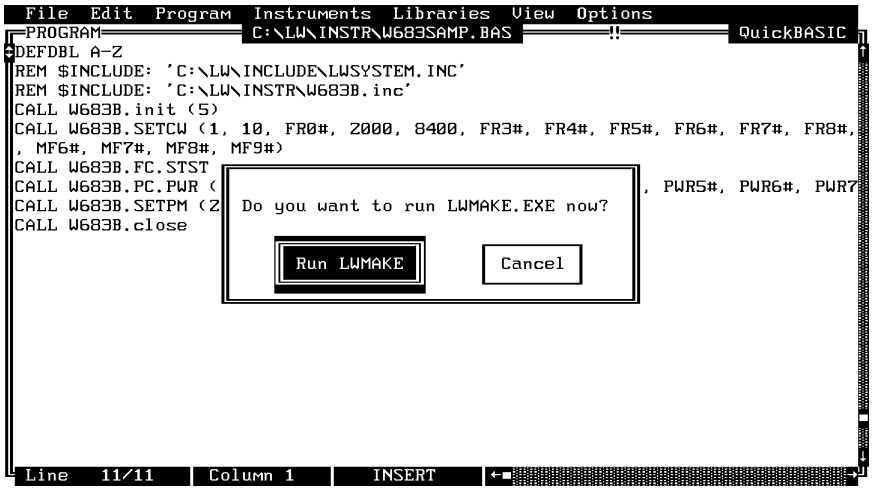

☐ Place yourself in the \LW\INSTR directory and select W683SAMP.BAS as you filename (below).



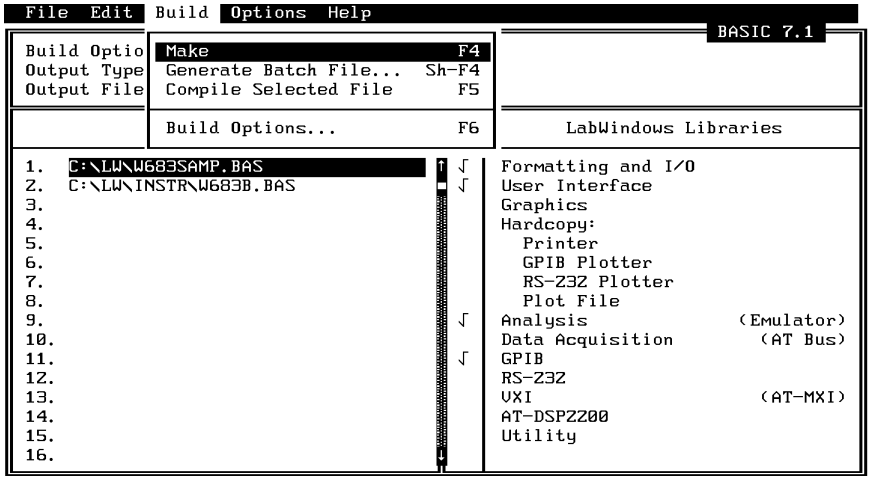

☐ Choose **Update** to save the changes to the file.



☐ Choose **Run LWMAKE,** in the next prompt.




☐ Move cursor to **Build** and choose **Make** from the pull-down menu. (**Note:** This assumes that the compile, link, and output paths have been previously defined under the LWMAKE Options menu.)




☐ At the conclusion of the processing that occurs next, the DOS executable file W683SAMP.EXE will appear in the subdirectory containing the like-named BASIC file. Press <ENTER> to return to the LWMAKE screen. To return to the LabWindows PROGRAM window, select **QUIT-Return to LabWindows** from the **File** menu.

☐ To check your handiwork, choose **DOS SHELL** from the **File** menu. At the ensuing DOS prompt, type `W683SAMP` to run the program.

# Section 3

# Driver References

# Section 3
# Driver References

## *Introduction*

This section provides descriptions of all of the 682XXB/683XXB Instrument Driver function calls. The functions are organized in alphabetical order.

## *Function Panel Structure*

The 682XXB/683XXB Labwindows Driver contains 21 panels that provide an intuitive method for coding instrument functions. Figure 3-1 shows the hierarchical structure of the functional panels. Table 3-1 lists these panels and shows the page number on which they are described.

**Table 3-1.** Functional Panels

| Panel Name | Page No. | Panel Name | Page No. |
|---|---|---|---|
| Alternate Sweep | 3-8 | Output Power Sweep | 3-32 |
| AM Modulation | 3-34 | Pulse Modulation | 3-46 |
| Close | 3-6 | Recall Setup | 3-50 |
| CW | 3-10 | Save Setup | 3-52 |
| Delta Sweep | 3-12 | Source Frequency Def | 3-20 |
| Frequency Modulation | 3-40 | Source Power Def | 3-22 |
| Frequency Setup | 3-38 | Special Sweep | 3-14 |
| Initialize | 3-18 | Start/Stop Sweep | 3-16 |
| Marker Selection | 3-44 | User Power Calibrate | 3-30 |
| Output Leveling | 3-24 | Utilities | 3-54 |
| Output Power | 3-28 | | |

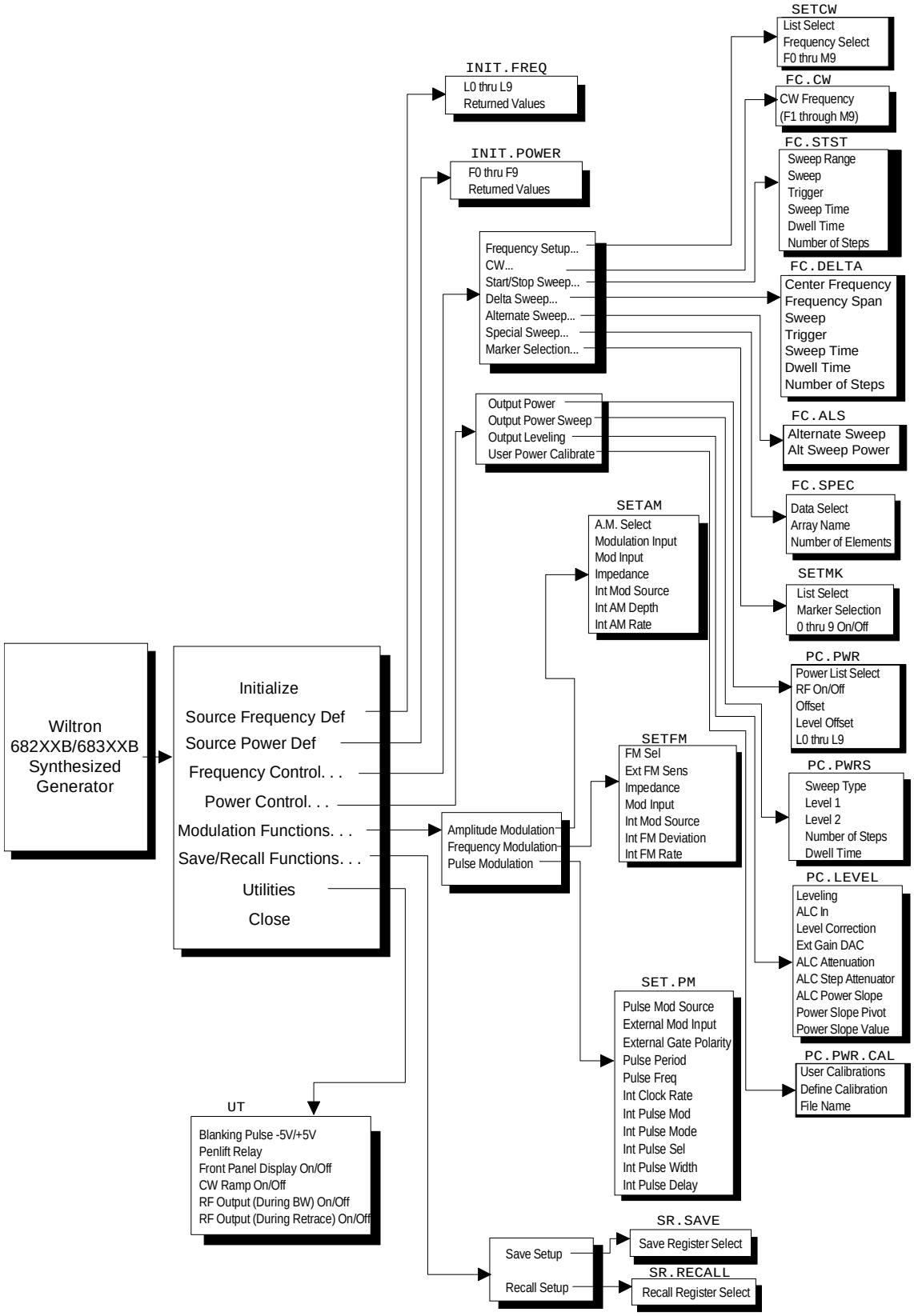
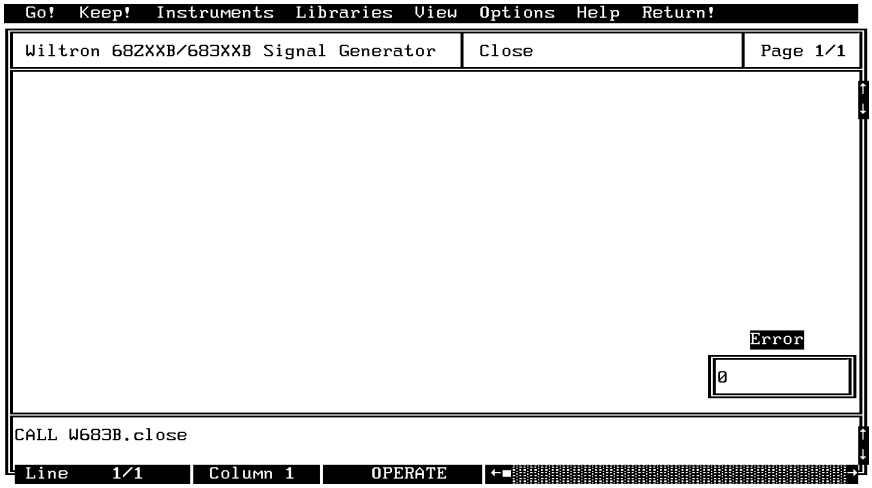



Figure 3-1. Function Panel Tree

# close

***Function Panel Name:*** Close

***Description:*** This function closes the GPIB communications with the 682XXB/683XXB.




***Controls:***

**None**

***Input Parameters:*** None.

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240.
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Close communications with the driver.
CALL W683B.close
```

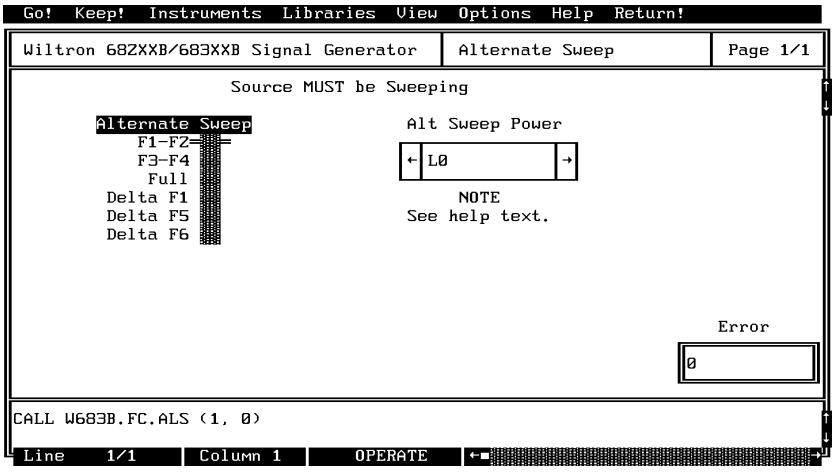
**Microsoft C:**

```
/* Close communications with the driver. */
W683B_close
```

# FC.ALS

***Function Panel Name:*** Alternate Sweep

***Description:*** This function lets users alternate an existing analog or phase-locked frequency sweep with another like-type sweep. The source must be sweeping prior to using this function or no action is taken.




***Controls:***

**Alternate Sweep:** Lets users select the sweep range that will alternate with the primary sweep: F1–F2, F3–F4, Full, Delta F5 (Δ F5), Delta F6 (Δ F6), Delta F1 (Δ F1). The 682XXB/683XXB must have a primary sweep selected and defined before an alternate sweep can be selected.

**Alt Sweep Power:** Selects the power parameter that is to be used for the Alt Sweep power level. The main sweep's power level is selected and defined in the Output Power (pc-pwr) panel.

**NOTE**

The main sweep and the alternate sweep may each have an independent power level associated with the frequency-sweep range. If you want the two sweeps to have different power levels, set the alternate sweep level in this panel and the main sweep power level in the Output Power panel. If you do not want to have different power levels, make the Alt Sweep Power parameter the same as the main sweep parameter. That is, if parameter L0 has been selected for the main sweep, then select L0 for the Alt Sweep also.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details | Range |
|---|---|---|---|---|
| ALTS% | Integer | Alternate Sweep | 1, F1–F2 (D)<br>2, Full Band<br>3, F3–F4<br>4, Delta F1<br>5, Delta F5<br>6, Delta F6 | N/A |
| S2L% | Integer | Alt Sweep Power | L0 (D) thru L9 | Model Dependent |

***Output Parameters:*** None.

***Error:*** LabWindows error codes 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Selects sweep to alternate with main sweep.
CALL W683B.FC.ALS (ALTS%, S2L%)
```

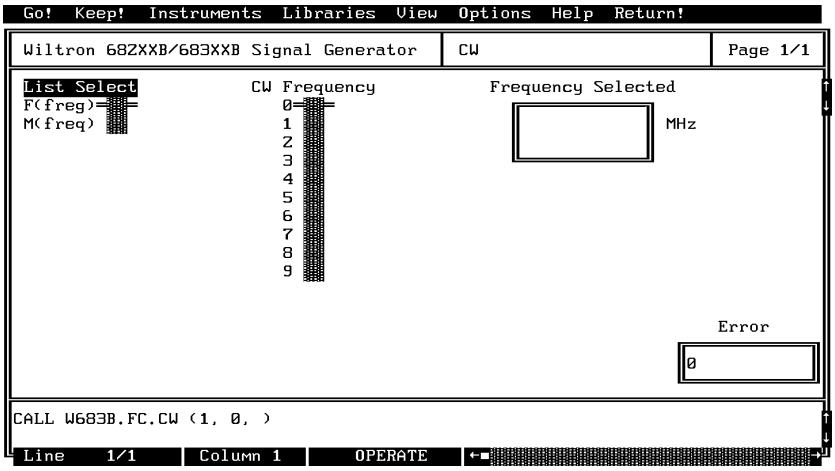
**Microsoft C:**

```
/*Selects sweep to alternate with main sweep. */
W683B_FC.ALS (ALTS,S2L)
```

# FC.CW

***Function Panel Name:*** CW

***Description:*** This function lets users select the desired CW output frequency from the values already stored in the 682XXB/683XXB frequency registers (F0 thru M9).




***Controls:***

**List Select:** Lets users select the F0–F9 frequency set or the M0–M9 frequency set.

**CW Frequency:** Lets users select CW output frequency from one of the twenty user-defined frequency registers: F0 through F9 or M0 through M9 (see `setcw` function, page 3-38).

**Frequency Selected:** Returns the CW frequency currently stored in the selected register, in MHz.

***Input Parameters:***

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| LS% | Integer | List Select | 1 = F(freq)<br>2 = M(freq) |
| FREQSEL% | Integer | The frequency of operation | 0–9 |

***Output Parameters:***

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| FSEL# | Double Precision | Frequency returned from 682XXB/683XXB. | |

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Select a CW frequency and value.
CALL W683B.FC.CW (LS%,FREQSEL%,FSEL#)
```

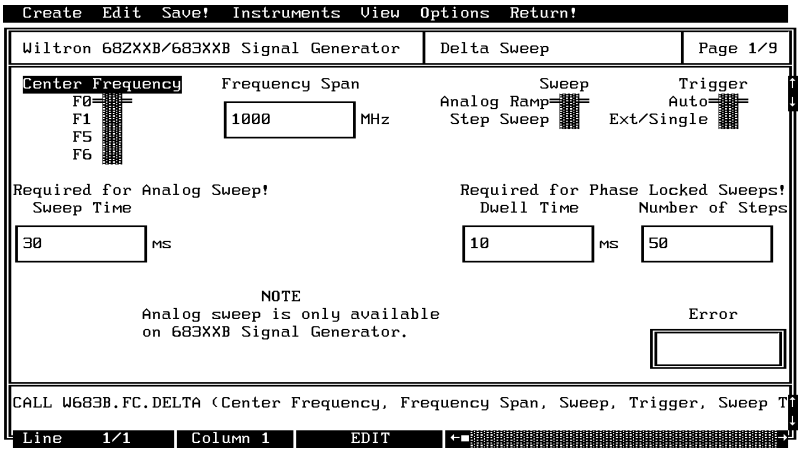
**Microsoft C:**

```
/* Select a CW frequency and value.*/
W683B_FC_CW (LS,FREQSEL,FSEL)
```

# FC.DELTA

***Function Panel Name:*** Delta Sweep

***Description:*** This function lets users set a center frequency and span for the sweep range.




***Controls:***

**Center Frequency:** Lets users select the center frequency F0, F1, F5, or F6 for a delta sweep.

**Frequency Span:** Lets users select a sweep width (span) in MHz. The 682XXB/683XXB will sweep upwards in frequency from *Center – (span/2) to Center + (span/2).*

**Sweep:** Lets users of 683XXB select either an analog or a phase-locked digitally stepped sweep. The 682XXA does not have analog sweep capability.

**Trigger:** Lets users select the triggering source for the delta sweep:

- **Auto:** Sweep is triggered automatically.
- **Ext/Single:** Sweep can be triggered manually by supplying a TTL pulse to pin 13 of the rear panel AUX I/O connector. The ability to launch a single sweep internally is currently inoperative.

**Sweep Time:** Lets users enter the time required for one analog sweep (683XXB only).

**Dwell Time:** Lets users enter the dwell time for a phase-locked sweep.

**Number of Steps:** Lets users enter the number of frequency steps for a phase-locked sweep.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details | Range |
|---|---|---|---|---|
| CFREQ% | Integer | Center Frequency | 1, F1 (D)<br>5, F5<br>6, F6 | N/A |
| FSPAN# | Double Precision | Frequency Span(MHz) | 1000 MHz (D) | (Analog Sweep)<br>1 MHz to full range.<br>(Digital Sweep)<br>1 kHz to full range. |
| SWEEP% | Integer | Sweep Type (683XXB only) | 0, Analog Ramp (D)<br>1, Step Sweep | N/A |
| TRIGG% | Integer | Trigger Conditions | 0, Auto (D)<br>1, Ext/Single (Currently inoperative) | N/A |
| SWEEPT# | Double Precision | Analog Sweep Time (683XXB) | 50 mSec (D) | 30 ms to 99 s. |
| DTIME# | Double Precision | PhaseLocked Dwell Time | 10 mSec (D) | 1 ms to 99 s. |
| NUMST% | Integer | Number of Steps For Phase Locked Sweeps | 50 (D) | 1 to 10,000 steps |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Selects delta sweep.
CALL W683B.FC.DELTA (CREQ%, FSPAN#, SWEEP%, TRIGG%, SWEEPT#,
DTIME#, NUMST%)
```

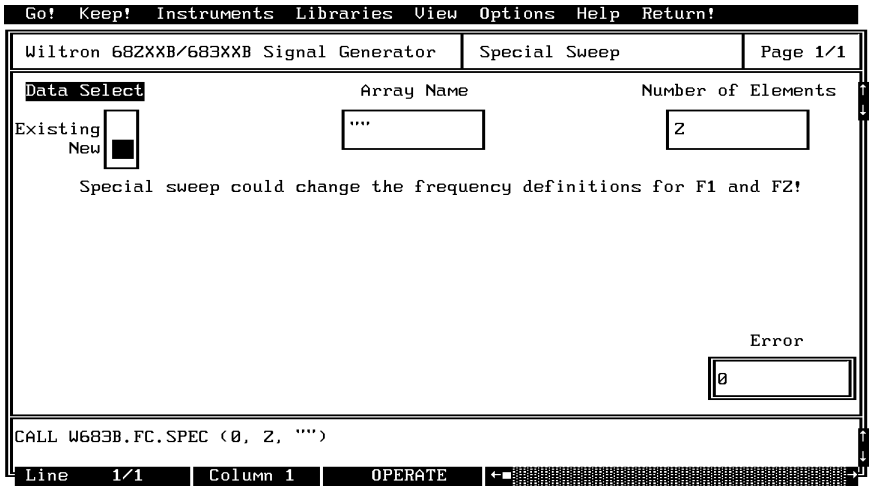
**Microsoft C:**

```
/* Selects delta sweep. */
W683B_FC_DELTA (CREQ, FSPAN, SWEEP, TRIGG, SWEEPT, DTIME, NUMST)
```

# FC.SPEC

***Function Panel Name:*** Special Sweep

***Description:*** This function lets users specify frequencies and the order that the frequency will be output. The operator can specify up to 1000 discrete frequencies.




***Controls:***

**Data Select:** Lets users select between new and existing *.wsw (disk file) data.

**Array Name:** Lets users enter the name of the data-array holding the new or existing data. If no array name is entered, the special sweep will still be generated. However, it will not be stored to disk in the current default directory.

**Number of Elements:** Enter the number of frequencies for a user-defined special sweep. The number of steps = (the number of frequencies –1).

***Input Parameters:*** Same as controls.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| DSEL% | Integer | Data select | 0 = New data file<br>1 = Existing data file |
| DPTS% | Integer | Number of Elements | Number of frequencies to be entered or recalled. |
| TEST$ | String | File Name | If no file name is entered the special sweep is still generated but not saved. The driver automatically appends a ".wsw" extension to the user-entered filename. The file is saved to the current default directory. |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240. If debug 3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Lets operator enter frequencies in Gigahertz for a user-de-
fined (special) sweep.
CALL W683B.FC.SPEC (DSEL%,DPTS%,TEST$)
```

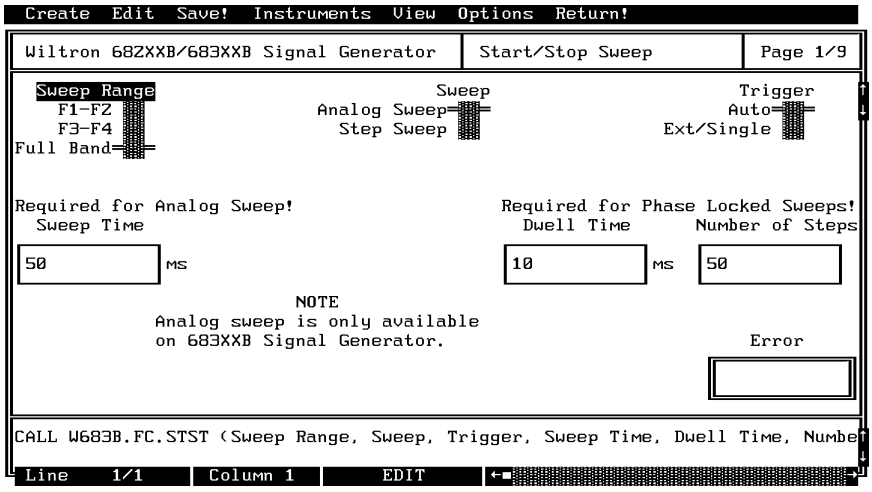
**Microsoft C:**

```
/* Lets operator enter frequencies in Gigahertz for a user-de-
fined  (special) sweep. */
W683B_FC_SPEC (DSEL,DPTS,TEST)
```

# FC.STST

***Function Panel Name:*** Start/Stop Sweep

***Description:*** This function lets users select the 682XXB/683XXB sweep range. The function uses the existing definitions for CW frequencies. The user has the opportunity to select the source trigger conditions and set sweep time — if in analog sweep. Or they can select dwell time and number of steps — if in phase locked sweep.




***Controls:***

**Sweep Range:** Lets users select the primary sweep range: F1–F2, F3–F4, or Full band.

**Sweep:** Lets users of 683XXB select either an analog or a phase-locked digitally stepped sweep. The 682XXA does not have analog sweep capability.

**Trigger:** Lets users select the triggering source for the primary sweep:

**Auto:** Sweep is triggered automatically by the 682XXB/683XXB.

**Ext/Single:** Sweep can be triggered manually by supplying a TTL pulse to pin 13 of the rear panel AUX I/O connector. The ability to launch a single sweep internally is currently inoperative.

**Sweep Time:** Lets users enter the time required for one analog sweep (no entry required for phase-locked sweep) (683XXB only).

**Dwell Time:** Lets users enter the dwell time between adjacent frequency steps for a phase-locked sweep (no entry required for analog sweep).

**Number of Steps:** Lets users enter the number of frequency steps for a phase-locked sweep (no entry required for analog sweep).

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details | Range |
|---|---|---|---|---|
| SWRANGE% | Integer | Sweep Range | 1,F1–F2<br>2, Full Range (D)<br>3, F3–F4 | N/A |
| SWEEP% | Integer | Sweep Type (683XXB only) | 0, Analog Ramp (D)<br>1, Step Sweep | N/A |
| TRIGG% | Integer | Trigger Conditions | 0, Auto (D)<br>1, Ext/Single (Currently inoperative) | N/A |
| SWEEPT# | Double Precision | Analog Sweep Time (683XXB only) | 30 mSec (D) | 30 ms to 99 s |
| DTIME# | Double Precision | PhaseLocked Dwell Time | 10 mSec (D) | 1 ms to 99 s. |
| NUMST% | Integer | Number of Steps For Phase Locked Sweeps | 50 (D) | 1 to 10,000 steps. |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Selects a primary (main) sweep.
CALL W683B.FC.STST (SWRANGE%, SWEEP%, TRIGG%, SWEEPT#, DTIME#,
NUMST%)
```

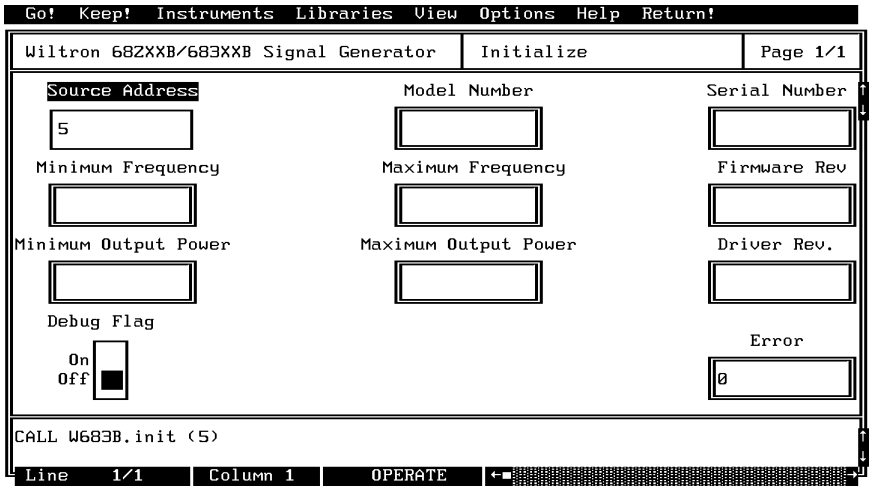
**Microsoft C:**

```
/* Selects a primary (main) sweep. */
W683B_FC_STST (SWRANGE, SWEEP, TRIGG, SWEEPT, DTIME, NUMST)
```

# init

***Function Panel Name:*** Initialize

***Description:*** This function opens the GPIB communication with the 682XXB/683XXB and sets it to its preset state. The device configuration must already be completed.




***Controls:***

**Source Address:** Lets users enter a GPIB address for the 682XXB/683XXB. The factory-set default address is 5.

**Model Number:** Displays the returned 682XXB/683XXB model number.

**Firmware Rev:** Displays the returned firmware revision level.

**Minimum Frequency:** Displays the returned minimum frequency for a full band sweep.

**Maximum Frequency:** Displays the returned maximum frequency for a full band sweep.

**Serial Number:** Displays the returned 682XXB/683XXB serial number.

**Minimum Output Power:** Displays the returned minimum output power to which the 682XXB/683XXB can be set.

**Maximum Output Power:** Displays the returned maximum output power to which the 682XXB/683XXB can be set.

**Driver Rev.:** Displays the driver's revision level.

**Debug Flag:** Lets users turn the Debug Utility on or off. This utility is discussed on page 2-8.

***Input Parameters:*** Same as controls.

| Variable Name | Variable Type | Description | Details | Range |
|---|---|---|---|---|
| ADD% | Integer | GPIB Address | 1–9, 5 (D) | 1 to 31 |

***Output Parameters:***

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| LOWF$ | String | Returned minimum frequency | 682XXB/683XXB OI string is parsed to supply values for the string variables shown in this table |
| HIGH$ | String | Returned maximum frequency | |
| MODEL$ | String | Returned model number | |
| SWRV$ | String | Returned software revision number | |
| MINPWR$ | String | Returned minimum power level | |
| GPWR$ | String | Returned guaranteed power level | |
| SSN$ | String | Returned instrument serial number | |

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Initializes the 682XXB/683XXB.
CALL W683B.init (ADD%)
```

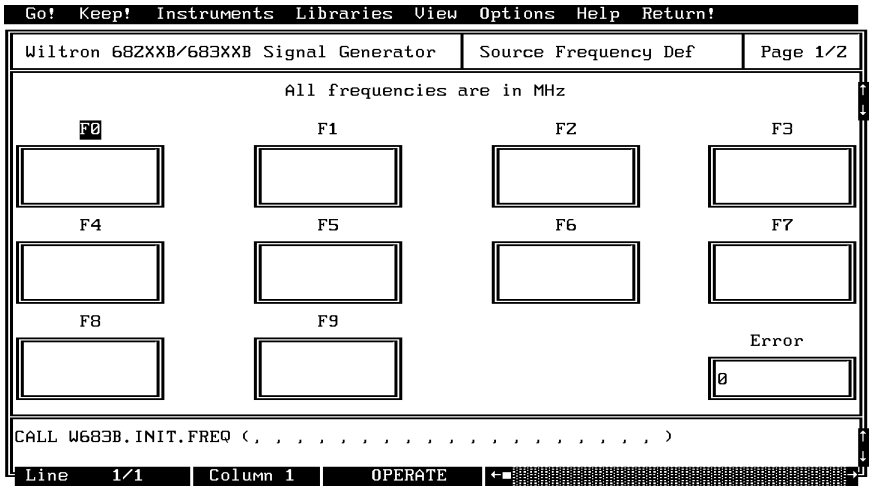
**Microsoft C:**

```
/* Initializes the 682XXB/683XXB. */
W683B_init (ADD)
```

# INIT.FREQ

***Function Panel Name:*** Source Frequency Def(inition)

***Description:*** This function returns the values of the frequencies (F0–F9, M0–M9) that are presently stored in the 682XXB/683XXB storage registers. All frequencies are returned in megahertz. There are no operator inputs.


Go! Keep! Instruments Libraries View Options Help Return!
Wiltron 68ZXXB/683XXB Signal Generator
Source Frequency Def
Page 1/2
All frequencies are in MHz
F0
F1
F2
F3
F4
F5
F6
F7
F8
F9
Error
0
CALL W683B.INIT.FREQ (, , , ,

***Input Parameters:*** None.

***Output Parameters:***

| Variable Name | Variable Type | Description |
|---|---|---|
| F0# thru F9# | Double Precision | CW frequencies active in the storage registers. |
| M0# thru M9# | Double Precision | CW frequencies active in the storage registers. |

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Get the value for up to twenty frequencies.
W683B.INIT.FREQ (F0#,F1#,F2#,F3#,F4#,F5#,F6#,F7#,F8#,F9#
M0#,M1#,M2#,M3#,M4#,M5#,M6#,M7#,M8#,M9#)
```

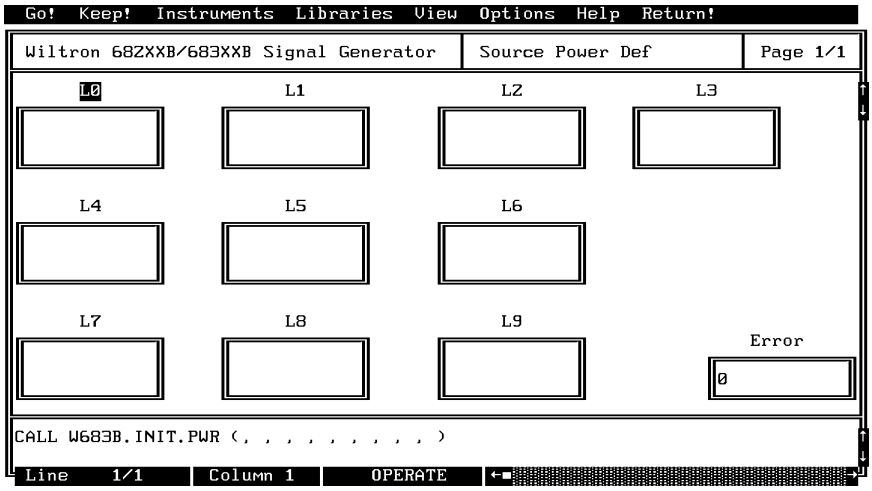
**Microsoft C:**

```
/* Get the value for up to nine frequencies. */
W683B_INIT_FREQ (F0,F1,F2,F3,F4,F5,F6,F7,F8,F9
M0,M1,M2,M3,M4,M5,M6,M7,M8,M9)
```

# INIT.PWR

***Function Panel Name:*** Source Power Def(inition)

***Description:*** This function returns the values of the power level values (L0–L9) that are presently stored in the 682XXB/683XXB storage registers. All values are returned in dBm. There are no operator inputs.




***Controls:***

**L0 – L9:** Displays the returned 682XXB/683XXB L0–L9 power level values.

***Input Parameters:*** None.

***Output Parameters:***

| Variable Name | Variable Type | Description |
|---|---|---|
| L0# thru L9# | Double Precision | Power level values active in the storage registers. |

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Get the value for up to nine power levels.
W683B.INIT.PWR (L0#,L1#,L2#,L3#,L4#,L5#,L6#,L7#,L8#,L9#)
```

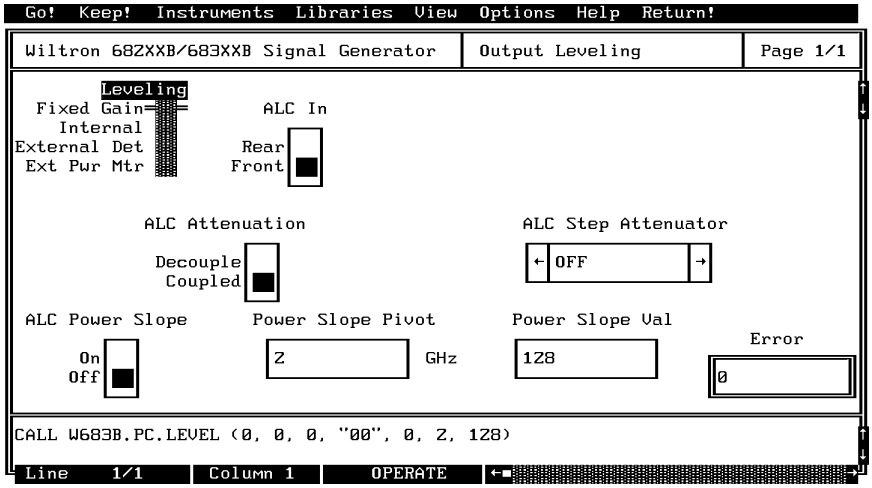
**Microsoft C:**

```
/* Get the value for up to nine frequencies. */
W683B_INIT_FREQ (L0,L1,L2,L3,L4,L5,L6,L7,L8,L9)
```

# PC.LEVEL

***Function Panel Name:*** Output Leveling

***Description:*** This function lets users select the power leveling mode(s).




***Controls:***

**Leveling** Lets users select the leveling mode for the output power.

**Fixed Gain:** ALC is disabled. The RF Level DAC and step attenuator (if installed) control the relative power level. Power is not detected at any point, and the absolute power level is uncalibrated.

**Internal:** Output power is leveled using an internal detector in the directional coupler to sample the output signal.

**External Det:** Output power is leveled using an external detector to sample the output signal.

**Ext Pwr Mtr:** Output power is leveled by using the "recorder voltage" from an external power meter. The 682XXB/683XXB is compatible with power meters having a ±1 V full scale analog output.

**ALC In:** Lets user select rear or front panel ALC input for external leveling.

| | |
|---|---|
| **ALC Step Attenuation:** | Couples or decouples the optional step attenuator from the ALC loop. |
| **ALC Step Attenuator:** | Pressing the right arrow turns the optional step attenuator on and increases it in 10 dB steps to 110 dB (decoupled mode only). |
| **ALC Power Slope:** | Lets users turn ALC power slope on or off. Power slope compensates for system, cable, and waveguide variations due to changes in frequency. |
| **Power Slope Pivot:** | Lets user enter a model-dependent frequency value for the power slope pivot point. Pivot point can be any frequency within the range of the instrument. |
| **Power Slope Value:** | Lets user enter a value for the ALC slope DAC of between 0 and 255. |

***Input Parameters:*** (D) is default setting

| Variable Name | Variable Type | Description | Details | Range |
|---|---|---|---|---|
| LEV% | Integer | Leveling | 0, Fixed Gain (D),<br>1, Internal<br>2, Ext Det<br>3, Ext Pwr Mtr | N/A |
| ALCCP% | Integer | ALC Attenuation | 0, Coupled (D)<br>1, Decouple | |
| ALCIN% | Integer | ALC In | 0, Front (D)<br>1, Rear | N/A |
| ATT$ | String | ALC Step Attenuator | “00" = Off (D)<br>”01" = 10 dB<br>“02" = 20 dB<br>”03" = 30 dB<br>“04" = 40 dB<br>”05" = 50 dB<br>“06" = 60 dB<br>”07" = 70 dB<br>“08" = 80 dB<br>”09" = 90 dB<br>“10" = 100 dB<br>”11" = 110 dB | N/A |
| ALCPS% | Integer | ALC Power Slope | 0, Off (D)<br>1, On | |
| PVT# | Double Precision | Power Slope Pivot | Default, 2 GHz | Model Dependent |
| VL% | Integer | Power Slope Value | Default, 128 | 1 to 255 |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Select the leveling mode.
CALL W683B.PC.LEVEL (LEV%,ALCCP%,ALCIN%,ATT$,ALCPS%,PVT#,VL%)
```

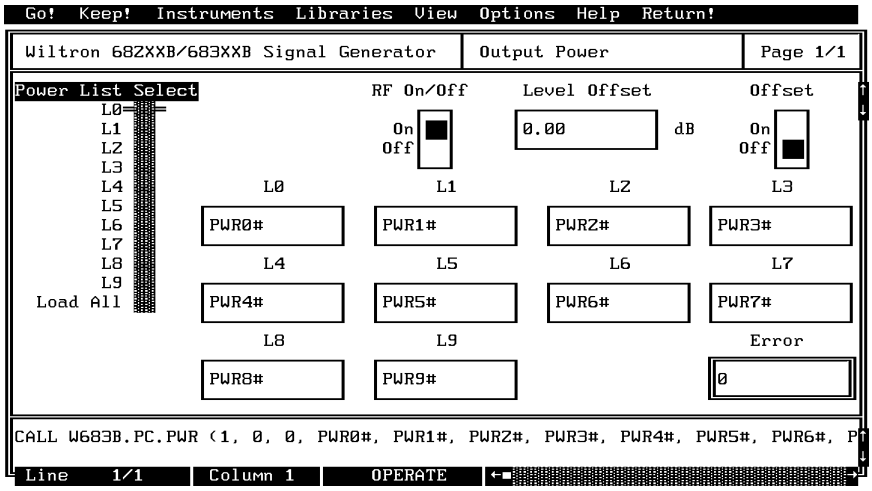
**Microsoft C:**

```
/* Select the leveling mode. */
W683B.PC_LEVEL (LEV,ALCCP,ALCIN,ATT,ALCPS,PVT,VL)
```

# PC.PWR

***Function Panel Name:*** Output Power

***Description:*** This function lets users set the output power level using an editable list of values. It also provides for turning the level offset function on or off and setting its value.




***Controls:***

**Power List Select:** Lets users select a power level parameter for loading in the 682XXB/683XXB.

**L0 – L9:** Opens the selected parameter and lets its value to be set and selected as the instrument output power level.

**Load All:** Opens all parameters and lets their value be set and the L9 parameter selected as the instrument output power level.

**RF On/Off:** Lets users turn the RF output power on or off.

**Offset:** Lets users turn the Level Offset function on or off.

**Level Offset:** Lets user enter a level offset value, when the function is turned on. This value compensates for a device in the output line that alters the level of the output power signal before being applied to the device-under-test.

**L0 thru L9:** Provide for entering power level values.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details | Range |
|---|---|---|---|---|
| PWRSEL% | Integer | Power List Select | 0 = L0 (D)<br>1 = L1<br>2 = L2<br>3 = L3<br>4 = L4<br>5 = L5<br>6 = L6<br>7 = L7<br>8 = L8<br>9 = L9<br>10 = Load All | N/A |
| OTP% | Integer | RF On/Off | 0 = Off<br>1 = On (D) | N/A |
| LEVOFF# | Double Precision | Level Offset | Value of Power Offset | +100 dB to –100 dB |
| LVLP% | Integer | Level Offset On/Off | 0, Off (D)<br>1, On | N/A |
| PWR0# thru PWR9# | Double Precision | L0 thru L9 | | Model Dependent. |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Set output power.
CALL W683B.PC.PWR
(PWRSEL%,OTP%,LEVOFF#,LVLP%,PWR0#,PWR1#,PWR2#,PWR3#,PWR4#,PWR5#,
PWR6#,PWR6#,PWR7#,PWR8#,PWR9#)
```

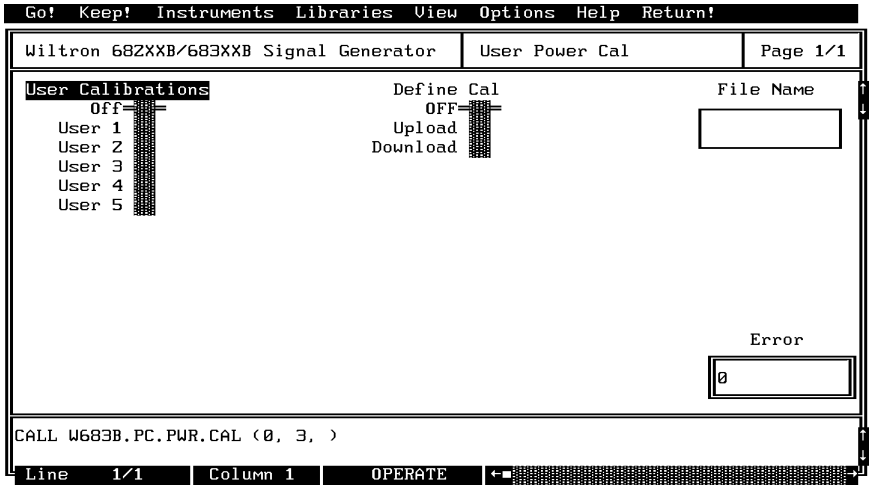
**Microsoft C:**

```
/* Set output power. */
W683B_PC_PWR (PWRSEL,OTP,LEVOFF,LVLP,PWR0,PWR1,PWR2,PWR3,
PWR4,PWR5,PWR6,PWR6,PWR7,PWR8,PWR9)
```

# PC.PWR.CAL

***Function Panel Name:*** User Power Calibrate

***Description:*** This function lets users enable and disable up to five internal power level correction tables created using the 682XXB/683XXB User Cal feature. This function also permits the user to upload and download the user cal data for all five table from an external computer. The purpose of the User Cal feature is to provide compensation for power level variations caused by external switching, amplifiers, couplers, and cables in the test setup.




***Controls:***

**User Calibrations:** Lets users activate a numbered, pre-defined, power level-flatness-correction table stored within the 682XXB/683XXB

**Define Calibration:**

**Off:** Lets users turn the level-correction feature off.

**Upload:** Uploads to disk all five of the level-correction tables stored in the 682XXB or 683XXB to the filename specified in the "File Name" field.

**Download:** Downloads data from the disk "File Name" file to the 682XXB or 683XXB that fills all five memory "User x" locations in the 682XXB or 683XXB.

***CAUTION***

Downloading data from the disk overwrites any data that was previously stored in these User x memory locations.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| UCAL% | Integer | User Calibration | 0 =Off (D)<br>1 = User 1<br>2 = User 2<br>2 = User 3<br>4 = User 4<br>5 = User 5 |
| DCAL% | Integer | Define Cal | 1 = Off(D)<br>2 = Upload<br>3 = Download |
| FAME$ | String | File Name | 8 characters maximum |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Set user cal feature power.
CALL W683B.PC.PWR.CAL (UCAL%,DCAL%,FAME$)
```

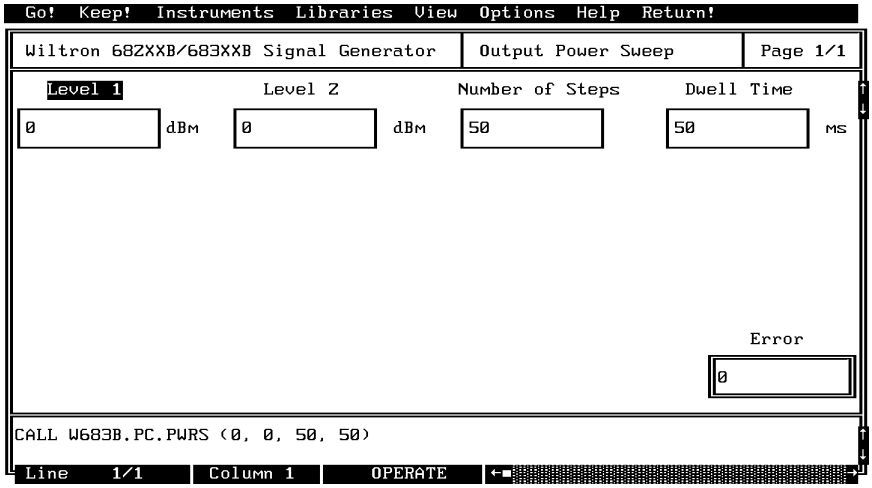
**Microsoft C:**

```
/* Set user cal feature power. */
W683B_PC_PWR_CAL (UCAL,DCAL,FAME)
```

# PC.PWRS

***Function Panel Name:*** Output Power Sweep

***Description:*** This function lets users set up the power sweep mode of operation.




***Controls:***

**Level 1 :** Lets users enter the setting in dBm for the start power level value. This is the start value for the power sweep.

**Level 2 :** Lets users enter the setting in dBm for the stop power level value. This is the stop value for the power sweep.

**NOTE**

Power sweep can be from high to low or low to high power levels. Therefore, the Level 1 (start) power could be higher or lower than the Level 2 (stop) power.

**Number of Steps:** Lets users enter the number of discrete power points (steps) at which the sweep will pause (as set using Dwell Time control).

**Dwell Time:** Lets users enter the length of time they want the power sweep to remain (dwell) at each step.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details | Range |
|---|---|---|---|---|
| LVL1# | Real | Level 1 Output Power | 0 dBm (D) | Depends on particular 682XXB or 683XXB model and whether optional Step Attenuator is fitted. Refer to 682XXB/683XXB Operation Manual for details. |
| LVL2# | Real | Level 2 Output Power | 0 dBm (D) | Same as above. |
| NUMST% | Integer | Number of Steps from Level 1 to Level 2 | 50 steps (D) | 0.01 to full power of instrument, which is model dependent. |
| DWT# | Real | Dwell Time At Each Step | 50 ms (D) | 1 ms to 99 s. |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Define the power sweep.
CALL W683B.PC.PWRS (LVL1#,LVL2#,NUMST%,DWT#)
```

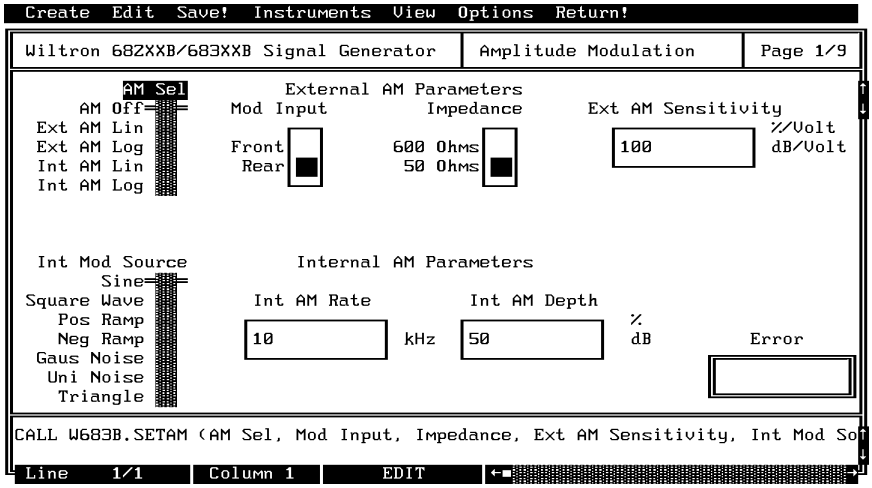
**Microsoft C:**

```
/* Define the power sweep. */
W683B_PC_PWRS (LVL1,LVL2,NUMST,DWT)
```

# SETAM

***Function Panel Name:*** Ampliltude Modulation

***Description:*** This function lets users set amplitude modulation (AM) parameters.




***Controls:***

**AMMod Sel:**

**AM Off:** Lets users turn AM on or off.

**Ext AM Lin:** Lets users select an external AM generator as the modulating source in a linear mode.

**Ext AM Log:** Lets users select an external AM generator as the modulating source in a log mode.

**Int AM Lin:** Lets users select the internal 682XXB/683XXB AM generator as the modulating source in a linear mode.

**Int AM Log:** Lets users select the internal 682XXB/683XXB AM generator as the modulating source in a log mode.

**Mod Input:** Lets uses select front or rear panel AM connectors.

**Impedance:** Lets users select the input inpedance of the front or rear panel AM connector to be 50Ω or 600Ω.

**Ext AM Sensitivity:** Lets users enter an AM sensitivity value in %/V for linear and dB/Volt for log selections on AM Mod Sel. The ranges are 1%/V to 100%/V for linear and 1dB/V to 25 dB/V for log.

**Int Mod Source**

| | |
|---|---|
| **Sine:** | Lets user select sinewave modulation. |
| **Square Wave:** | Lets user select square wave modulation. |
| **Pos Ramp:** | Lets user select positive-going ramp modulation. |
| **Neg Ramp:** | Lets user select negative-going ramp modulation. |
| **Gaus Noise:** | Lets user select Gaussian noise modulation. |
| **Uni Noise:** | Lets user select uniform noise modulation. |
| **Triangle:** | Lets user select triangle-wave modulation. |
| **Int AM Rate:** | Lets user enter a modulation rate. The range is 0.1 MHz to 1 MHz for sinewave and 0.1 Hz to 100 kHz for square, triangle, and ramp waveforms. |
| **Int AM Depth:** | Lets user enter a value for amplitude modulation depth. The depth ranges are 1% to 100% for linear and 1dB to 25 dB for log. |

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| AMSEL% | Integer | AM Sel | 0 - AM Off (Default)<br>1, Ext AM Lin<br>2, Ext AM Log<br>7, Int AM Lin<br>8, Int AM Log |
| AMIN% | Integer | Mod Input | 3, Front<br>4, Rear (D) |
| ZSEL% | Integer | Impedance | 6, 600 Ohms<br>5, 50 Ohms |
| AMSENS# | Double Precision | Ext AM Sensitivity | 100 %/V (D) |
| INTMOD% | Integer | Int Mod Source | 1, Sine (D)<br>2, Square Wave<br>3, Pos Ramp<br>4, Neg Ramp<br>5 Gaus Noise<br>6, Uni Noise<br>7, Triangle |
| INTDEPTH# | Double Precision | Int AM Depth | 50 % (D) |
| INTRATE# | Double Precision | Int AM Rate | 10 kHz (D) |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Set modulation values for AM.
CALL W683B.SETAM (AMSEL%,AMIN%,ZSEL%,AMSENS#,INTMOD%,
INTDEPTH#,INTRATE#)
```

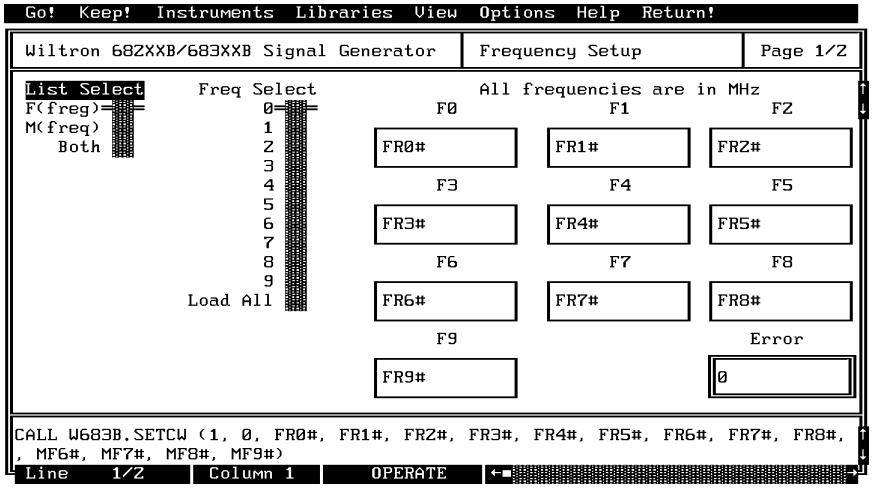
**Microsoft C:**

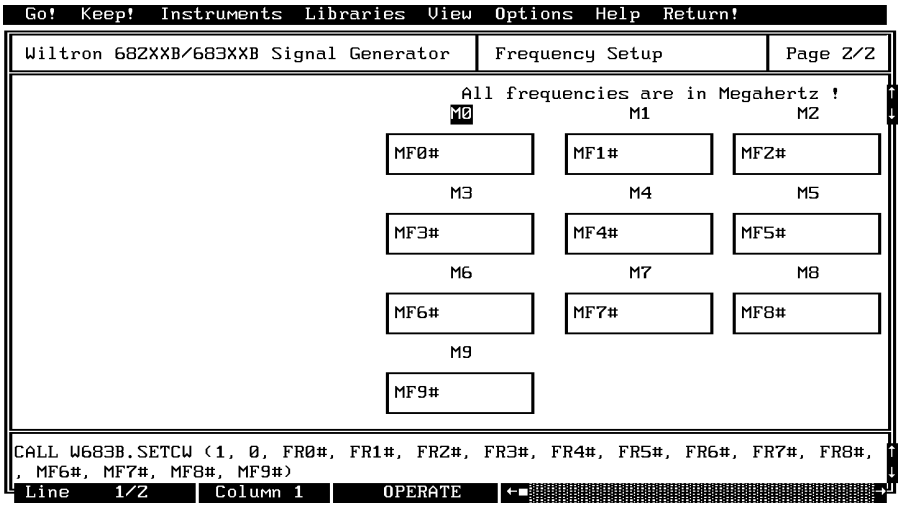
```
/* Set modulation values for AM. */
W683B_SETAM (AMSEL,AMIN,ZSEL,AMSENS,INTMOD,INTDEPTH,INTRATE)
```

# SETCW

***Function Panel Name:*** Frequency Setup

***Description:*** This function lets users define frequencies for any of the twenty 682XXB/683XXB storage registers.




***Controls:***

**List Select:** Lets users select the F0-F9 set, M0-M9 set, or both sets for assignment of frequency values. If "Both" is selected, all defined frequencies will be loaded and the 682XXB/683XXB will be set to output the M-frequency selection (0–9 on Frequency Select control).

**Frequency Select:** Lets users select any single frequency storage location, or to select all twenty at one time. If more than one frequency is to be defined, the switch must be set to the Load All position. *Press the <Page Down> key to call up page 2 and the Mx set for assigning frequency definitions*

**F0 thru F9**
**M0 thru M9:** Let users enter a frequency value in MHz for storage in the applicable location (register).

***Input Parameters:***

| Variable Name | Variable Type | Description |
|---|---|---|
| FREQSEL% | Integer | Selects Frequencies F0–F9 or M0–M9, where 10 loads all frequencies. If any other value, it selectively loads that value into the corresponding register. |
| FRx# | Real | Frequency to be loaded into the storage registers. |
| MFx# | Real | Frequency to be loaded into the storage registers. |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Select a cw frequency storage location and load frequency
value.
CALL W683B.SETCW (FREQSEL%,FR0#,FR1#,FR2#,FR3#,FR4#,FR5#,
FR6#,FR7#,FR8#, FR9#,MF0#,MF1#,MF2#,MF3#,MF4#,MF5#,MF6#,MF7#,
MF8#,MF9#)
```

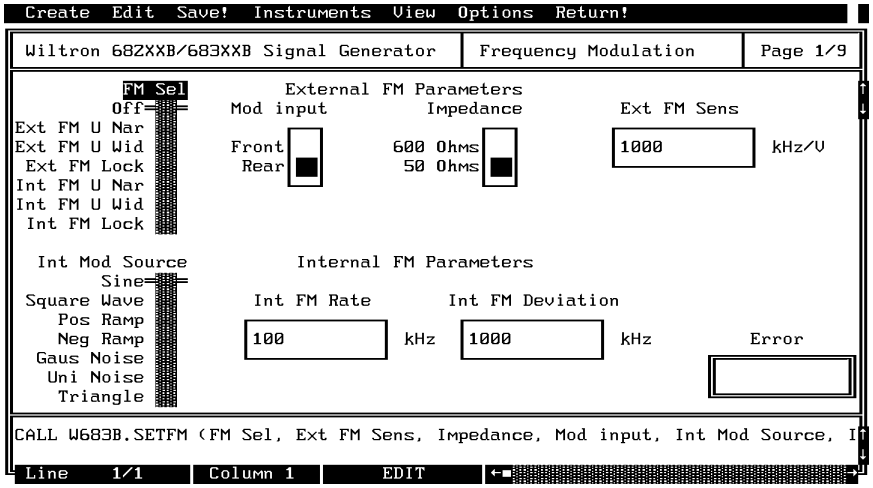
**Microsoft C:**

```
/* Select a cw frequency storage location and load frequency
value. */
W683B_SETCW (FREQSEL,FR0,FR1,FR2,FR3,FR4,FR5,FR6,FR7,FR8,
FR9,FR0,MF1,MF2,MF3,MF4,MF5,MF6,MF7,MF8,MF9)
```

# SETFM

***Function Panel Name:*** Frequency Modulation

***Description:*** This function lets users set modulation parameters for frequency modulation.




***Controls:***

**FM Sel:**

**Off:** Lets users turn FM on or off.

**Ext FM U Nar:** Lets users select an external source in the Unlocked, Narrow frequency modulation mode.

**Ext FM U Wid:** Lets users select an external source in the Unlocked, Wide frequency modulation mode.

**Ext FM Lock** Lets users select an external source in the Locked frequency modulation mode.

**Int FM U Nar:** Lets users select the internal source in the Unlocked, Narrow frequency modulation mode.

**Int FM U Wid:** Lets users select the internal source in the Unlocked, Wide frequency modulation mode.

**Int FM Lock** Lets users select the internal source in the Locked, frequency modulation mode.

**Mod Input:** Lets uses select front or rear panel FM connectors.

**Impedance:** Lets users select the input inpedance of the front or rear panel FM connector to be 50Ω or 600Ω.

**Ext FM Sensitivity:** Lets users enter an FM sensitivity value for the external source. Sensitivity range is ±1 kHz/V to ±20 MHz/V for Locked and Unlocked Narrow FM modes and ±100 kHz/V to ±100 MHz/V for Unlocked Wide FM mode.

**Int Mod Source**

**Sine:** Lets user select sinewave modulation.

**Square Wave:** Lets user select square wave modulation.

**Pos Ramp:** Lets user select positive-going ramp modulation.

**Neg Ramp:** Lets user select negative-going ramp modulation.

**Gaus Noise:** Lets user select Gaussian noise modulation.

**Uni Noise:** Lets user select uniform noise modulation.

**Triangle:** Lets user select triangle-wave modulation.

**Int FM Rate:** Lets user enter a modulation rate. The range is 0.1 MHz to 1 MHz for sine wave and 0.1 Hz to 100 kHz for square, triangle, and ramp waveforms.

**Int FM Deviation:** Lets user enter a value for amplitude modulation depth. The depth ranges are ±1 kHz to ±20 MHz for Locked and Unlocked Narrow FM modes and ±100 kHz to ±100 MHz for Unlocked Wide FM mode.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| FON% | Integer | FM Sel | 0, Off (D)<br>1, Ext FM U Nar<br>2, Ext FM U Wid<br>3, Ext FM Lock<br>7, Int FM U Nar<br>8, Int FM U Wid<br>9, Int FM Lock |
| SENS# | Real | FM Sensitivity in kHz/Volt | 1 MHz/V (D) |
| IMPED% | Integer | Impedance | 5 = 50 Ohms (D)<br>6 = 600 Ohms |
| MODIN% | Integer | Modulation Input | 3 = Front<br>4 = Rear (D) |
| INTMOD% | Integer | Int Mod Source | 1, Sine (D)<br>2, Square<br>3, Pos Ramp<br>4, Neg Ramp<br>5, Gaus Noise<br>6, Uni Noise<br>7, Triangle |
| INTDEV# | Double Precision | Int FM Deviation | 1 MHz (D) |
| INTRATE# | Double Precision | Int FM Rate | 100 kHz (D) |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Set modulation values for FM.
CALL W683B.SETFM (FON%,SENS#,IMPED%,MODIN%,INTMOD%,INTDEV#,
INTRATE#)
```

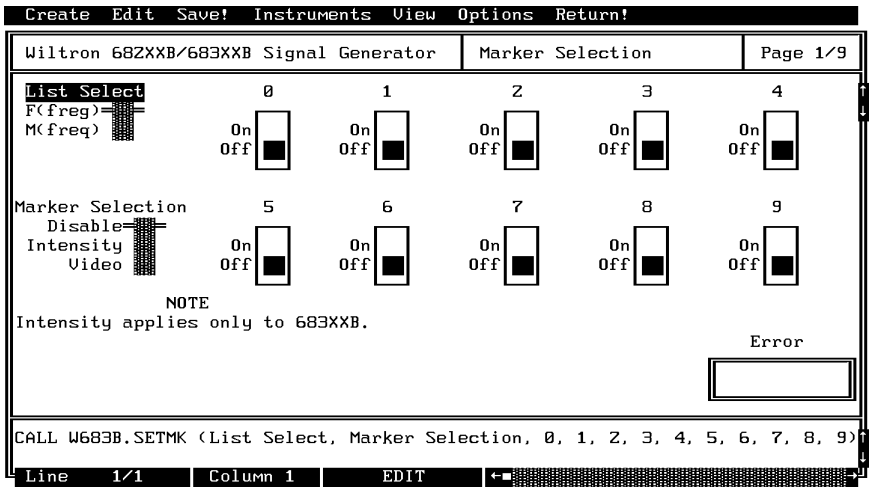
**Microsoft C:**

```
/* Set modulation values for FM. */
W683B_SETFM (FON,SENS,IMPED,MODIN,INTMOD,INTDEV,INTRATE)
```

# SETMK

***Function Panel Name:*** Marker Selection

***Description:*** This function lets users set video and intensity markers.




***Controls:***

**List Select:** Lets users select the F0–F1 or M0–M1 frequency set.

**Marker Selection:** Lets users disable markers or select between Intensity and Video markers.

**Disable:** Disables the frequency markers.

**Intensity:** Enables an intensified dot to mark the frequency, provided the selected marker frequency is within the selected frequency range. (Only available in analog sweep mode for 683XXB.)

**Video:** Enables a video pulse to mark the frequency, provided the selected marker frequency is within the selected frequency range.

**0 thru 9:** Turns marker on or off at the frequency stored in locations F0 thru F9 or M0 thru M9, based on the position of the List Select control.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| MKSEL% | Integer | Marker Select | 0, Disable (D)<br>1, Intensity<br>2, Video |
| F0% Thru F9% | Integer | Marker:0–9 | 0, Off (D)<br>1, On |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Set markers.
CALL W683B.SETMK (MKSEL%,F1%,F2%,F3%,F4%,F5%,F6%,F7%,F8%,F9%)
```

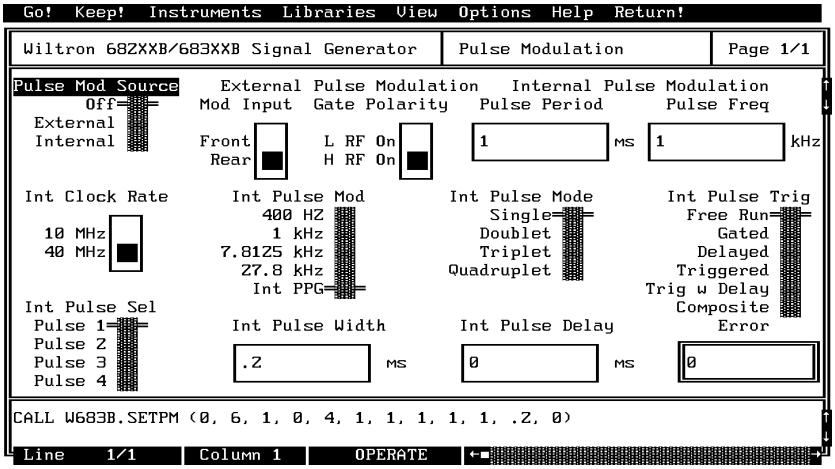
**Microsoft C:**

```
/* Set markers. */
W683B_SETMK (MKSEL,F1,F2,F3,F4,F5,F6,F7,F8,F9)
```

# SETPM

***Function Panel Name:*** Pulse Modulation

***Description:*** This function lets users set square wave modulation parameters.




***Controls:***

**Pulse Mod Source:**

**Off:** Lets users turn Pulse modulation off.

**External:** Lets users select an external source for pulse modulation.

**Internal:** Lets users select the initernal source for the pulse modulation.

**Mod Input:** Lets uses select front or rear panel Pulse Mod connectors.

**Gate Polarity:** Lets users select the polarity of the signal that turns the RF on: Low (L RF) or High (H RF).

**Pulse Period:** Lets users enter a pulse period (in milliseconds) for the internal pulse generator.

**Pulse Freq** Lets users enter a pulse repetition frequency (PRF) (in KHz) for the internal pulse generator.

**NOTE**

Since a pulse's period and frequency are reciprocal values, you should make an entry on only one of these fields. If entries are made in

both, the 682XXB or 683XXB will respond to the last value received.

**Int Clock Rate:** Lets users select between 10 MHz and 40 MHz rates.

**10 MHz:** This rate provides 100 ns pulses at lower PRFs.

**40 MHz:** This rate provides the highest resolution (25 ns) pulses and allows the highest PRF.

**Int Pulse Mod:** Lets the user select between internal pulse modulation values of 400 Hz, 1 kHz, 7.8125 kHz, 27.8 kHz, or the internal pulse-period generator (PPG).

**Int Pulse Mode:** Lets users select an internal pulse mode.

**Single:** This mode provides modulation using single pulses. Pulse rate and delay can be set using the Int Pulse Sel control, below.

**Doublet:** This mode provides modulation using a 2-pulse doublet set. Pulse rate and delay for each pulse in the doublet can be set using the Int Pulse Sel control, below.

**Triplet:** This mode provides modulation using a 3-pulse triplet set. Pulse rate and delay for each pulse in the triplet can be set using the Int Pulse Sel control, below.

**Quadruplet:** This mode provides modulation using a 4-pulse quadruplet set. Pulse rate and delay for each pulse in the quadruplet can be set using the Int Pulse Sel control, below.

**Int Pulse Trig:**

**Free Run:** Pulse generator free runs; D1 pulse delay not available.

**Gated:** Pulse generator gated on by external pulse. Single pulse mode only.

**Delayed:** Pulse generator free runs; D1 pulse delay is available.

**Triggered:** Pulse generator triggered by external trigger; D1 pulse delay not available.

**Trig w Delay:** Pulse generator triggered by external trigger; D1 pulse delay is available.

**Composite:** An external pulse is summed with the internal pulse to pulse modulate the output signal.

**NOTE**

For a detailed description of the internal pulse triggering modes, refer to the 682XXB/683XXB Operation Manual, ANRITSU Part No. 10370-10284.

**Int Pulse Sel:** Lets users select pulses in a set (single, doublet, triplet, quadruplet) for individual modification.

**Int Pulse Width:** Lets users enter a pulse width for the pulse selected using the Int Pulse Sel control.

**Int Pulse Delay:** Lets users enter a pulse delay for the pulse selected using the Int Pulse Sel control.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| MODS% | Integer | Pulse Mod Source | 0, Off (D)<br>1, External<br>2, Internal |
| MODIN% | Integer | Mod Input | 5, Front<br>6, Rear (D) |
| XPG% | Integer | Gate Polarity | 0, L RF On<br>1, H RF On (D) |
| IPRF% | Integer | Int Pulse Mod | 0, Int PPG (D)<br>1, 400 Hz<br>2, 1 kHz<br>3, 7.8125 kHz<br>4, 27.8 kHz |
| INTCLCK% | Integer | Int Clock Rate | 1, 10 MHz<br>4, 40 MHz (D) |
| INTPMODE% | Integer | Int Pulse Mode | 1, Single (D)<br>2, Doublet<br>3, Triplet<br>4, Quadruplet |
| INTPTRIG% | Integer | Int Pulse Trig | 1, Free Run (D)<br>2, Gated<br>3, Delayed<br>4, Triggered<br>5, Trig w Delay<br>6, Composite |
| INPSEL% | Integer | Int Pulse Sel | 1, Pulse 1(D)<br>2, Pulse 2<br>3, Pulse 3<br>4, Pulse 4 |

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| PPER# | Double Precision | Pulse Period | 1 ms (D) |
| PFREQ# | Double Precision | Pulse Frequency | 1 ms (D) |
| PW# | Double Precision | Internal Pulse Width | 0.2 ms (D) |
| DLY# | Double Precision | Internal Pulse Delay | 0 ms (D) |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Set pulse modulation parameters.
CALL W683B.SETPM (MODS%,MODIN%,XPG%,IPRF%,INTCLCK%,
  INTPMODE%,INTPTRIG%,INPSEL%,PPER#,PFREQ#,PW#,DLY#)
```

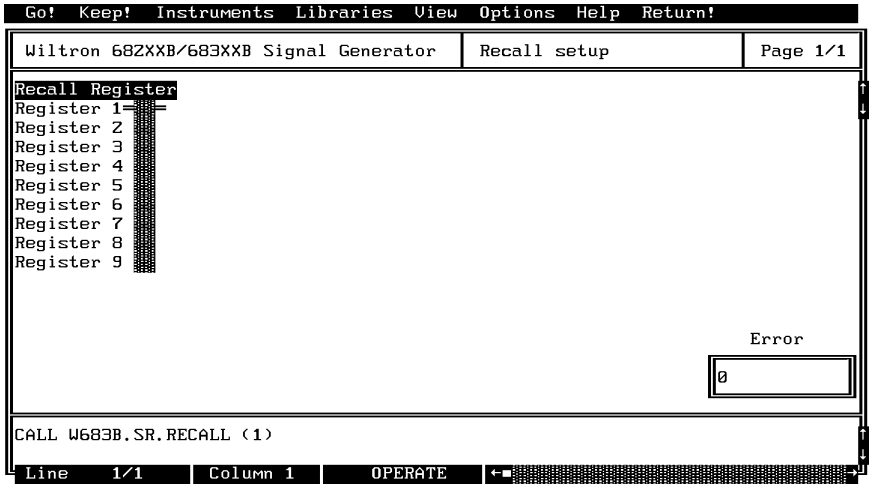
**Microsoft C:**

```
/* Set pulse modulation parameters */
W683B_SETPM (MODS,MODIN,XPG,IPRF,INTCLCK%,INTPMODE,INTPTRIG,
INPSEL,PPER,PFREQ,PW,DLY)
```

# SR.RECALL

***Function Panel Name:*** Recall Setup

***Description:*** This function lets users recall an existing source setup from the 682XXB/683XXB internal memory.




***Controls:***

**Recall Register:** Lets users select the storage register containing the setup they wish to recall.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| REG% | Integer | Storage Register, 1 Thru 9 | 1, (D) |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Recall stored setups.
CALL W683B.SR.RECALL (REG%)
```

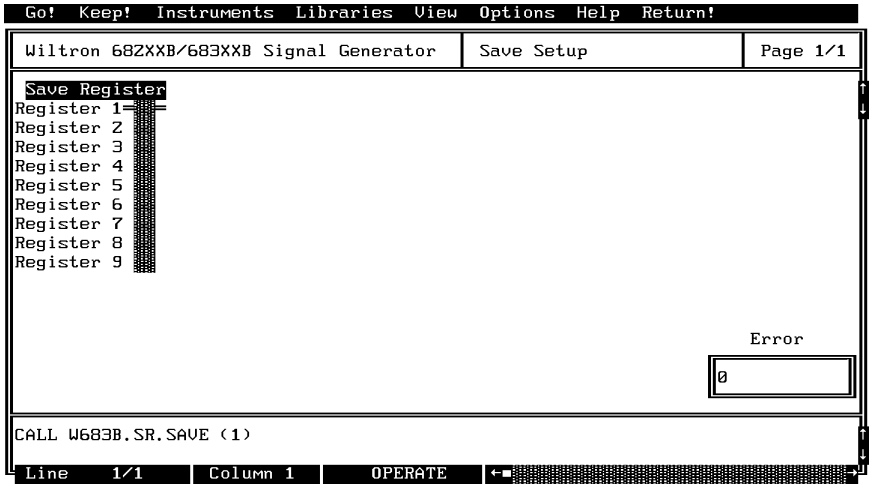
**Microsoft C:**

```
/* Recall stored setups. */
W683B_SR.RECALL (REG)
```

# SR.SAVE

***Function Panel Name:*** Save Setup

***Description:*** This function lets users save an existing setup to the 682XXB/683XXB internal memory.




***Controls:***

**Save Register:** Lets users select a storage register for storing a setup they may wish to recall at a later time.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| REG% | Integer | Save Register, 1 Thru 9 | 1, (D) |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Recall stored setups
CALL W683B.SR.SAVE (REG%)
```

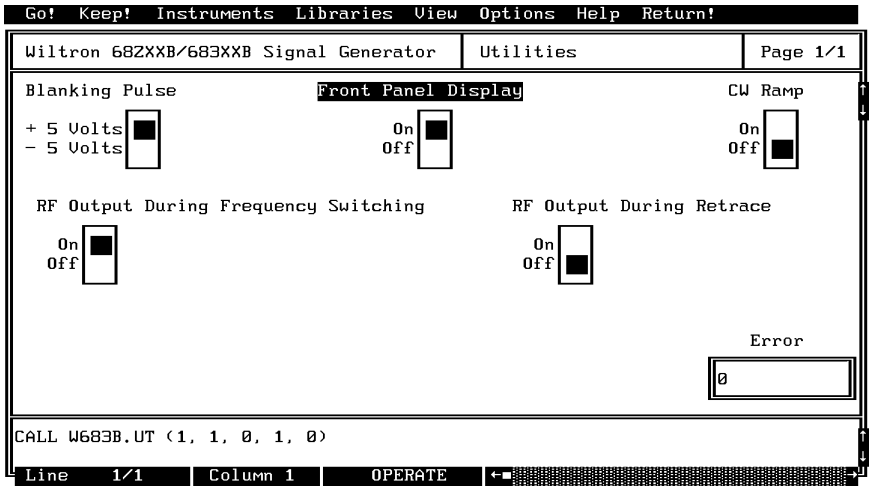
**Microsoft C:**

```
/* Recall stored setups */
W683B_SR_SAVE (REG)
```

# UT

***Function Panel Name:*** Utilities

***Description:*** This function lets users define many additional functions typically used for advanced applications. It also controls the 682XXB/683XXB RF output during frequency switching and retrace.




***Controls:***

**Blanking Pulse:** Lets users select rear panel supplied blanking pulse to be ±5 volts during sweep retrace.

**Front Panel Disp:** Lets users turn the front panel displays on or off.

**CW Ramp:** Lets users turn the CW Ramp function on or off. This function causes a 0–10 volt horizontal ramp to be supplied to the rear panel HORIZONTAL OUTPUT connector during CW operations.

**RF Output – During Frequency Switching:** Lets users turn the RF output on or off during band-switching.

**RF Output – During Retrace:** Lets users turn the RF output on or off during sweep retracing.

***Input Parameters:*** (D) is default setting.

| Variable Name | Variable Type | Description | Details |
|---|---|---|---|
| BLNKP% | Integer | Blanking Pulse | 0, –5 Volt<br>1, +5 Volt (D) |
| RFFS% | Integer | RF During Bandswitching | 0, Off<br>1, On (D) |
| RFDR% | Integer | RF During Retrace | 0, Off (D)<br>1, 0n |
| FPD% | Integer | Front Panel Display | 0, Off<br>1, On (D) |
| CWR% | Integer | Rear Panel CW Ramp Out | 0, Off (D)<br>1, On |

***Output Parameters:*** None.

***Error:*** LabWindows error codes only 220 to 240
3XX Instrument Specific (Page 2-8)

***Program Examples:***

**Quick BASIC:**

```
REM Set blanking, CW ramp, RF output (during freq switching and
retrace), penlift relay, front panel display, and hi resolution
values.
CALL W683B.UT (BLNKP%,RFFS%,RFDR%,FPD%,CWR%)
```

**Microsoft C:**

```
/* Set blanking, CW ramp, RF output (during freq switching and
retrace), penlift relay, front panel display, and hi resolution
values. */
W683B_UT (BLNKP,RFFS,RFDR,FPD,CWR)
```