

Microvision Application Note:

Suite Setup *(Automatic Test Software)*

Additional info. :
SS200 Series Manuals

VERSION 1.0
DECEMBER 1, 1999

<i>About (Purpose of) this Document</i>	3
<i>System Requirements:</i>	3
<i>Suite Setup - Purpose</i>	3
<i>Getting Started</i>	4
Menu Structure and Mode Changing	4
Menu Structure	4
Setting SS230 Mode	5
<i>Test Suite Functions</i>	6
Test Conditions	7
Test Selection	8
Timing Files.....	8
Screen Test Selection.....	8
Points to Test (not yet implemented)	9
Point Test Selection	9
Display Setup (not yet implemented)	9
Run Tests	9
Erase Log File.....	9
Erase Selected tests.....	9
Run Selected Tests.....	9
Log Test Results	9
View Test Results	10
<i>Suite Setup 'Setup'</i>	10
Entering DUT report information	10
Report Customization	10
Transit Grid setup	11
<i>Log Files and Data</i>	11
File Formats	11
File Locations	12
Test Numbers	12
Test Locations	13
Interpreting Log File Data	14
<i>Tests Requiring Hardware Change</i>	13
Adding Data to existing Log File	13
Using Incorrect Hardware	13

About (Purpose of) this Document

This document describes how to use the Suite Setup function of the SS200 series systems. It is intended to be used in conjunction with the SS200 series manuals and will not serve as a single source for operation of the software.

System Requirements:

Requires an SS200/210 system or an SS220 or both (SS230) Running Versions 1.61 or higher. Depending on which system you have, some functions may not be available for inclusion into your test suite. Before using the Suite Setup, you should have a good working knowledge of the SS200 series systems. This document is applicable for either the DOS or the Windows version of the program.

Suite Setup - Purpose

The purpose of the suite setup function is to allow the operator to build a custom set of tests that are tailored to his specific needs. Suite set makes available all the functions contained in the SS200 series systems to be included into a signal automated test. The result is a comprehensive and repeatable testing sequence that will generate a report of the tests and greatly reducing the amount of operator interaction while increasing accuracy and throughput.

The tests are included into the test suite by clicking on the function name from a list of available functions. Once all the desired functions are checked, the test is started and no more operator intervention is required. The report is generated in a simple ASCII format but is also available in a log file for importing into data base programs such as Excel.

Getting Started

Menu Structure and Mode Changing

Purpose: To describe recent changes to the SS200 series software to include menu options and mode selection.

Description: The appearance and organization of the SS200 series software has been modified since the last printing of the manual. The changes have been made in an effort to combine several system options into a single software package to increase the ease of operation. Because the SS200 series offers several different camera configurations to choose from, the new menu structure and mode change will allow the operator to change quickly between cameras for various measurements with out requiring a restart of system software. As the transformation occurs into a single software package, the features present in other packages will be incorporated with the ultimate discontinuation of these other separate packages. The following will describe the operation of the new menu/mode structure.

Menu Structure

The menu structure groups related functions onto a single menu. Each menu contains tests that would commonly be used in conjunction with one another. For example, the functions on the first menu all relate to measurements that are normally made on CRT displays, Menu 2 relates to Flat panel displays, menu 3 lists all the various automated test suites, and menu 4 is used for custom applications. Below is a list of the currently available functions on each menu.

SS200:OMM	MENU 2	MENU 3	MENU 4
SPOT ANALYSIS	FPDM FULL SCREEN	SUITE SETUP	CUSTOM 1
LINE ANALYSIS	FPDM BOX-PATTERN	ISO 9241-3&8: TC095	CUSTOM 2
RASTER ANALYSIS	FPDM UNIFORMITY	NIDL CRTDM	CUSTOM 3
GEOMETRY ANALYSIS	FPDM VIEWING ANGLE	ISO 13406-2	CUSTOM 4
CONVERGENCE ANALYSIS			CUSTOM 5
LUMINANCE ANALYSIS			CUSTOM 6
COLOR ANALYSIS			CUSTOM 7
			DEMO FNS
			SYSTEM CONTROL
MENU 2	MENU 3	MENU 4	MU DIAGNOSTICS

Menu 1 - This menu is primarily used for CRT analysis. It contains all the standard SS200 functions as previous versions of software. Functions in this menu are generally associated with OMM mode.

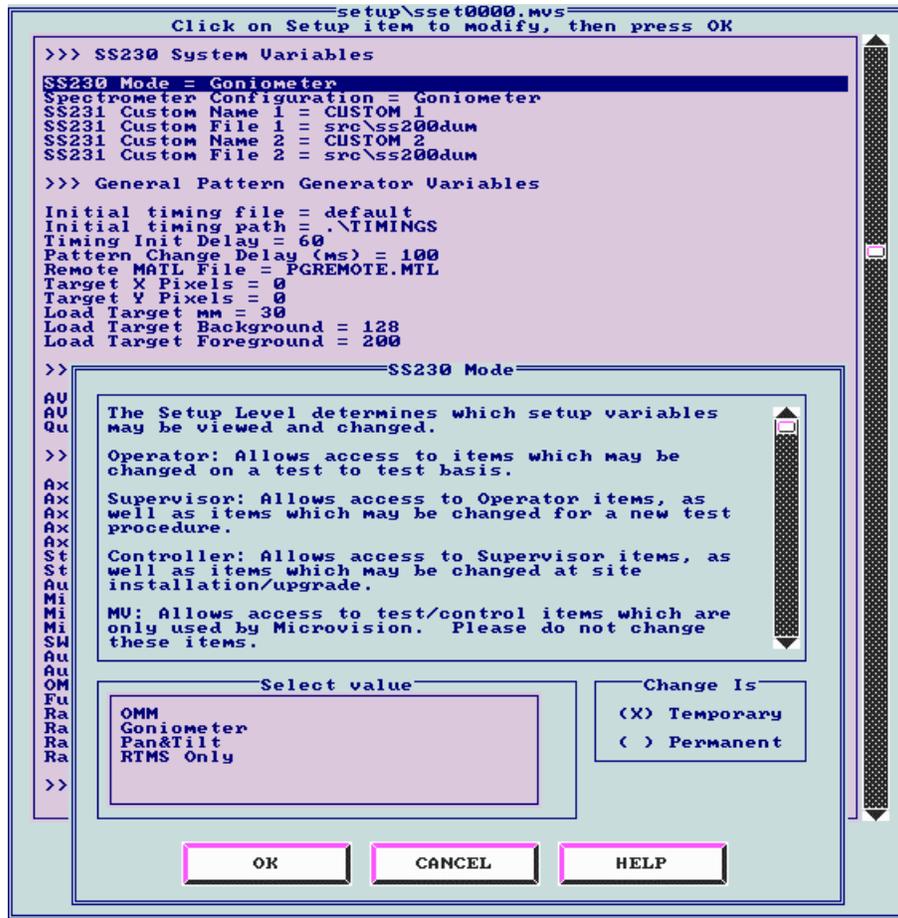
Menu 2 - This menu is primarily used for FPD analysis. The functions here currently consist of FPDM measurements but will later include the functionality of the SS220 software package. Functions in this menu will require Goniometer mode although many can be performed in OMM mode.

Menu 3 - This menu contains the various automated test suites that are available. Test suites will use a variety of different modes and some may require a mode change in order complete the analysis.

Menu 4 - This menu is a used defined menu for placement of custom MATL programs. MATL (Microvision Automatic Test Language) is a C like language used to write test routines. Although accessible to everyone, this menu is basically reserved for the MATL programmer.

Setting SS230 Mode

The enabling of the software (changing modes) is done within the setup function of the main system screen.



Selecting the operating mode of the system from the main setup menu. Mode changes may be necessary to perform tests that require different hardware.

Operating Modes:

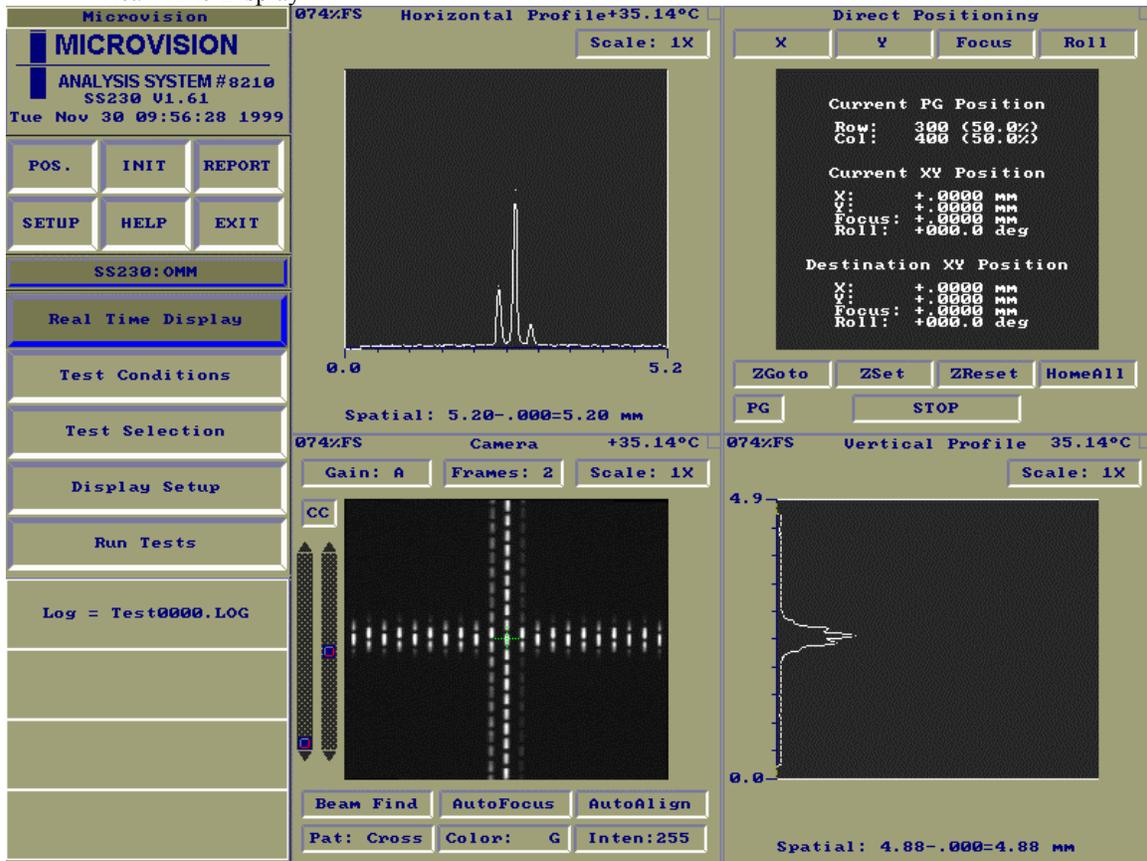
- **OMM** - CCD camera and spectrometer and optional RTM. (SS210) Select this mode to run Temporal or Response time measurements if you have an OMM with RTM installed.
- **Goniometer** - goniometric assembly with bench top spectrometer. Select this mode to run the majority of VESA tests.
- **Pan & Tilt** - not yet available
- **RTMS Only** - Stand alone RTM unit. Select this mode if you are going to use a stand-alone RTM module to make Temporal or Response Time measurements.

Test Suite Functions



After clicking on the Suite Setup button (Menu 3), the above function menu will appear. These are the functions that will enable you to create and execute your automated test suite. The following sections will now describe each of the functions.

Real Time Display



The Real Time Display function is similar to those found elsewhere in the system. It allows the operator to view and adjust various conditions prior to testing.

Test Conditions

Use Conditions of Test Defaults

Test Condition Questionaire:
 Use Test Condition Defaults?
 Normally, you should respond "NO" to be prompted for the test conditions for this test.
 You may respond "YES" to use the conditions established by the setup values, or by the previous test (if one exists).
 You should respond "YES" only to repeat a test, or to test with the same conditions, but a new pattern generator timing.

Select value

No
Yes

The tests conditions function allows the operator to enter information about the UUT (DUT) prior to testing. Manufacturer, model number, and serial number are examples of the type of information to be entered. To use the system defaults or those from a previous test, answer Yes to the prompt shown above.

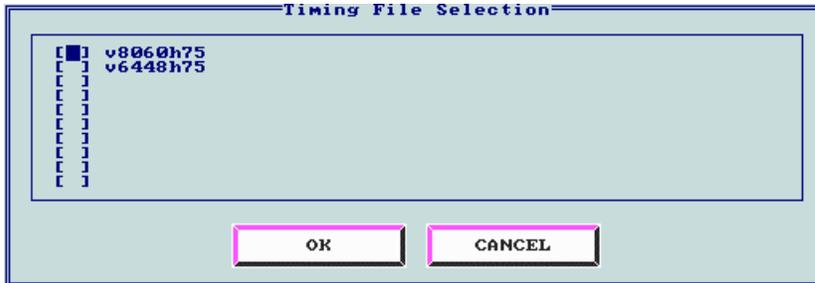
The default or current values can be seen (and altered) in the setup function. By answering No, the user will be prompted with a series of questions regarding the UUT. This information will be included in the final test report. See the section Suite Setup 'Setup' for a list of the UUT information to be entered.

Test Selection



The test selection function first prompts the operator with a list for choosing the area of the automated test suite to be configured. Each of the individual areas shown above will be described in the next four sections.

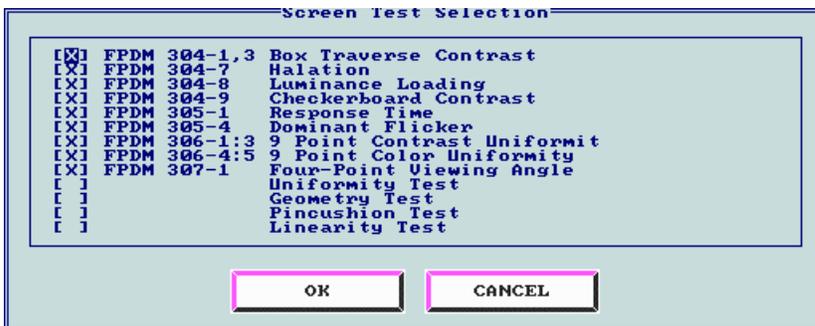
Timing Files



The Suite Setup can be configured to collect data using several timing files. The Suite will first load the selected timing file and then execute the entire suite. After completion, the next checked timing file will be loaded and the entire suite will be run again. Note that after each timing file is loaded, the system must re-initialize before the suite is run. If no timing files are selected, the current timing will be used and a restart is not necessary.

Note: Initially, the timing file list will be empty. The operator must manually enter the timing file list into the main setup. The timing file names are those found in the timings directory, c:\mv\timings or those listed when a 'Timing' initialization is performed.

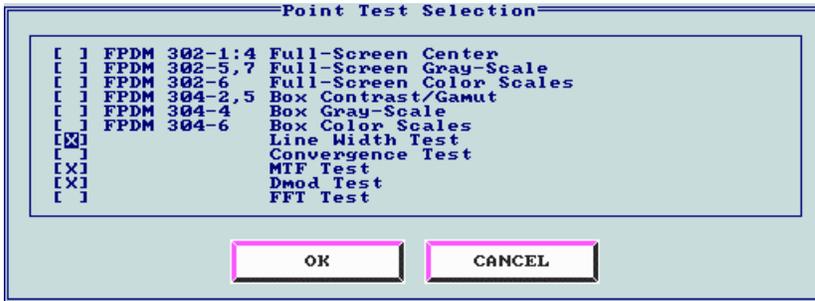
Screen Test Selection



Screen tests are those standard system functions that may require stage motion and data to be collected at several locations, or tests that can only be made at one location. For example, the linearity function requires that the stage move to a predetermined set of coordinates while collecting the linearity data, or the Halation function which requires the camera positioned at the center of the UUT.

Points to Test (not yet implemented)

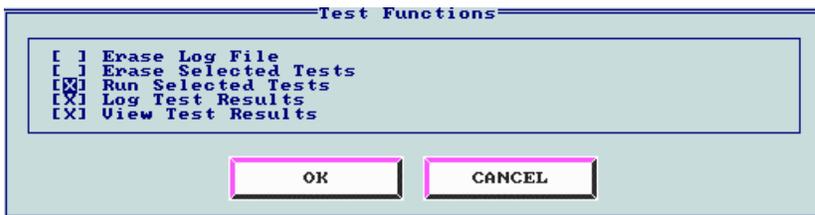
Point Test Selection



Point Tests are those functions, which can be run at any location on the UUT and whose measurement results are not dependent on results collected at other locations. They do not have a predetermined set of coordinates like many of the screen functions do. Because of their location independence, they can be run at any location on the UUT. We can then establish a grid of test locations where we would like each of the Point Tests to be executed. See the Suite Setup 'Setup' section for information on establishing a test grid.

Display Setup (not yet implemented)

Run Tests



The Run Tests function enables the operator to control the logging of data and report generation.

Erase Log File

Checking this box will delete the existing log file prior to starting the test suite.

Erase Selected tests

Checking this box will delete only the selected tests (from screen tests and points tests) from the log file. This is used when results from a particular test are not to be included in a report.

Run Selected Tests

Checking this box will run the tests selected in screen or point tests. If a test has already been run (log file created) and only the report is to be generated, then this box would not be checked.

Log Test Results

Checking this box will log the results from each test into the log file. This is used if tests are to be run but the data collected from each test is not to be saved.

View Test Results

Checking this box will display the test results on the screen after data is collected. This will require operator interaction before more testing can be done. This function is also used when test results are to be displayed (generated) from a previously existing log file but no actual data is to be collected.

Suite Setup 'Setup'

```
Use Conditions of Test Default = No
DUT Manufacturer =
DUT Model =
DUT Serial Number =
DUT Revision Level =
DUT Technology = AM-TFT-LCD
DUT Configuration = Laptop
DUT Active Area (hor/mm) = 0
DUT Active Area (ver/mm) = 0
DUT Dimensions (hor/mm) = 0
DUT Dimensions (ver/mm) = 0
DUT Dimensions (depth/mm) = 0
DUT Mass (weight) (kg) = 0
DUT Design Viewing Direction = Perpendi
DUT Design Viewing Distance (m = 400
DUT Bits Per Color = 6R,6G,6B
DUT Color Bits[0] = 18
DUT Gray Levels[0] = 64
DUT Total Colors =
DUT Power Source = internal
Testing Organization =
Test Location =
Test Operator =
Test Sequence/Run Number =
Test Unit Warm Up Time = 0
Pattern Generator % drive = 100
Target offset from corner (%) = 10

*** Conditions of Test variables above

*** Report Customization ***
Report Header Form = suiterh.frm
Timing Change Form = suitetc.frm
Save Screen Images? = Yes

*** Transit grid setup ***
Multiple Points = Use Grid
Grid Horz. Lines = 3
Grid Vert. Lines = 3
Grid X Pixel Origin = 0
Grid Y Pixel Origin = 0
Grid X Pixel Size = -100
Grid Y Pixel Size = -100
Grid Control = 0

*** OPTION VARIABLES ***
Log = Test0000.LOG
```

The Suite Setup 'Setup' screen is shown above. Clicking the 'Setup' button once you are in the Suite setup menu accesses this screen.

Entering DUT report information

The first part of the setup table shows the current settings of the 'Test conditions'. These are the values that may have been entered through the use of the 'Test Conditions' function. The Test conditions can also be entered or changed directly here. You may find this a more convenient method of entering information. The information contained here will also be included in the final report.

Report Customization

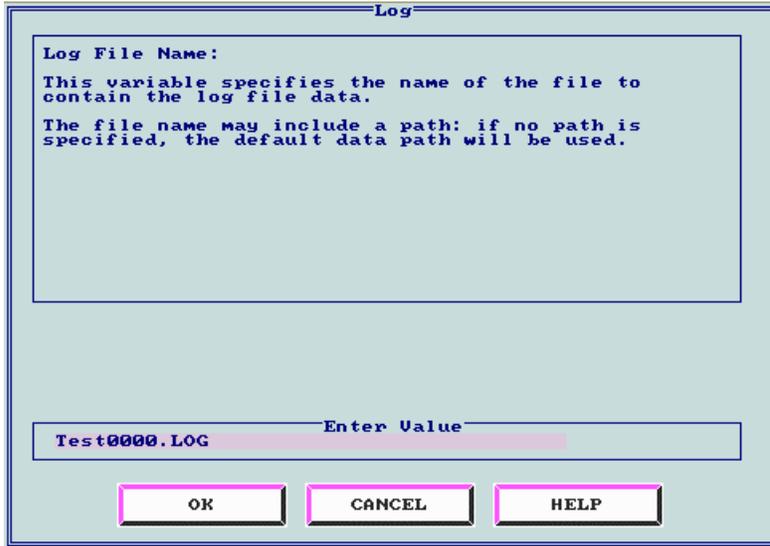
Test reports are generated at the completion of each test. Preceding the report data is header information created by each of the 'form' files (. frm) listed above, namely, the Report Header Form and the Timing Change Form. The report headed includes the DUT information entered into the setup and the timing change form list information regarding the current timing file. (KHz, pixels, etc.). The look and content can be modified by the user but will require a working knowledge of MATL programming. A copy of the MATL manual can be found on Microvision's FTP site ftp.microvsn.com.

Each of the resulting display screens (screen dumps) can also be saved to disk if the 'Save Screen Images' variable is set to 'Yes'. See the section on 'Log files' for more information on file locations and naming conventions.

Transit Grid setup

The transit grid defines the locations on the screen that the Point Tests will be performed. The points are located on an imaginary grid of N x M dimensions. Setup variables determine the number of grid lines to be used as well as the origin and size of the grid. The grid locations will only be collected if the 'Multiple Points' variable is set to 'Use Grid'.

Log Files and Data



Log files are generated each time the Suite Setup is run. The log files are comma delimited ASCII files and are to be used with various data base programs such as Excel. The reports that are generated after running a test suite are created from the data contained in this file.

File Formats

Test0000.log - comma delimited log files with the entire test results and information.

Test0000.rep - report file generated from the information contained in the log file.

Screen Dump files - Screen dump files have the following naming convention - Fxxxx-yy.pcx
Where xxxx is the location of the test and yy is the test number. See 'Test Numbers and Test Locations' for more information on test numbering.

Example of data in log file:

```
.  
.
Log.Unixunits#0,0,10,"u"
Log.Uniyunits#0,0,10,"v"
Log.Uniphot#0,0,20,"fL"
Log.Geostats#0,0,12,3,3,9,303,228,34.9471,51.2604,0.496335,0.364521...
.  
.
```

The data above is typical of what you would find in log file. See the section 'Interpreting Log File Data' for more information about the variables.

File Locations

Files created during the execution of a Test Suite, such as log files, reports files, and screen dumps, are saved in a subdirectory under 'c:\mv\data' with the same name as the log file. For example, the above files would be saved in the directory:

C:\mv\data\test0000

Test Numbers



Each function that can be selected in Suite Setup has a unique function number and is used to identify which test has been run. The number will appear in reports and on screen dumps. The above picture shows what will be displayed in the upper right corner of the screen when test number 10 is executed at location 00. Below is a list of all the available functions and their corresponding function numbers.

Point Test numbers

00 - FPDM 302-1:4 Full-Screen Center
01 - FPDM 302-5,7 Full-Screen Gray-Scale
02 - FPDM 302-6 Full-Screen Color Scales
11 - FPDM 304-2,5 Box Contrast/Gamut
12 - FPDM 304-4 Box Gray-Scale
13 - FPDM 304-6 Box Color Scales
28 - Line Width Test
29 - Convergence Test
30 - MTF Test
31 - Dmod Test
32 - FFT Test

Screen Test numbers

10 - FPDM 304-1,3 Box Traverse Contrast
14 - FPDM 304-7 Halation
15 - FPDM 304-8 Luminance Loading
16 - FPDM 304-9 Checkerboard Contrast
17 - FPDM 305-1 Response Time
20 - FPDM 305-4 Dominant Flicker
21 - FPDM 306-1:3 9 Point Contrast Uniformity
22 - FPDM 306-4:5 9 Point Color Uniformity
25 - FPDM 307-1 Four-Point Viewing Angle
33 - Uniformity Test
34 - Geometry Test
35 - Pincushion Test
36 - Linearity Test

Test Locations

Each location that is tested is also assigned a location number. The number ranges from 0 to the number test points in the grid minus 1. For example, a 3 x 3 grid will have test locations 0 through 8. Note that data is collected in a serpentine fashion and the location number is incremented at each location. The resultant location numbers are:

0 1 2
5 4 3
6 7 8

Tests Requiring Hardware Change

Each of the tests listed in the Suite Setup has a particular type of camera associated with it. For example, the 'Line Width' measurement will require the OMM module in order to use the 2 dimensional CCD detector where as the 'Four Point Viewing Angle' function will require the Goniometer for off axis measurements.

Adding Data to existing Log File

To accommodate test suites that require the use of more than one camera type, log files are handled such that data can be added after a hardware change has been made. By running the initial data collection with one set of hardware and then changing to a new camera and running again, data can be added to the log file to include results from more than one camera.

Using Incorrect Hardware

It is not necessary to check and then uncheck specific tests because a camera change has been made. The Suite Setup will automatically determine if the appropriate camera is installed for a specific test and then notify you at the end of test that a hardware change may be necessary. Be aware that some tests can be run by either camera type, such as 'Color Gamut' which requires the use of the spectrometer. Both the Goniometric camera and the SS210 OMM camera contain spectrometers. In this case, you may wish to 'uncheck' some of the tests that use the spectrometer to prevent them from being run again.

Interpreting Log File Data

Below is a description of the log file variable format followed by a list of variables.

Variable, index, elements, value...

Variable - Label associated with data set. used for Vlookup(Excel).

Index - currently unused.

Elements - number of array elements (values) to follow

Value - first array value,second array value,.....,last array value

Example:

Log.LineWidth#0,0,11,5,10,50,0.984362,0.729249,0.442112,0.651215,0.565125,0.310275,52.65,49

index = 0.

There are 11 values in array. 5 is first,then 10,50... last 49

The make up of the elements are as follows:

Log.LineWidth#0[11];

```
//index,# of elements,  
//linewidth%0,linewidth%1,linewidth%2,  
//HorzLw0,HorzLw1,HorzLw2,  
//VertLw0,VertLw1,VertLw2,  
//HorzPeakLum,VertPeakLum (CCD value)
```

When multiple LineWidth values are collected while traversing grid,
the '#' will increment. Log.LineWidth#1,Log.LineWidth#2,....

Log Array Value Definitions: Organization of array values.

All the Log arrays start with the index followed by the number
of elements in the array followed by the values.

Variable list:

```

Log.FSCGamutData [18]; // [R,G,B,W,Blk] [Lum,x,y],cct,colorspace,Fill
Factor

Log.BoxGamutData [18]; // [R,G,B,W,Blk] [Lum,x,y],cct,colorspace,Fill
Factor

Log.FSCGammaInf [16]; // [W,R,G,B] [Steps, minlevel, maxlevel,
FillFactor]
Log.FSCGammaDrv [256]; // [W,R,G,B] [64 max levels]drive
level,Lum,x,y,CCT]
Log.FSCGammaLum [256]; // [W,R,G,B] [64 max levels]luminance cd/m2
Log.FSCGammaClx [256]; // [W,R,G,B] [64 max levels]x
Log.FSCGammaCly [256]; // [W,R,G,B] [64 max levels]y
Log.FSCGammaCCT [256]; // [W,R,G,B] [64 max levels]CCT
Log.FSCGammaRes [24]; // [W,R,G,B] [gamma(val,cc,mini,maxi),
deltauv(max,avg)]

Log.BoxGammaInf [16]; // [W,R,G,B] [Steps, minlevel, maxlevel,
FillFactor]
Log.BoxGammaDrv [256]; // [W,R,G,B] [64 max levels]drive
level,Lum,x,y,CCT]
Log.BoxGammaLum [256]; // [W,R,G,B] [64 max levels]luminance cd/m2
Log.BoxGammaClx [256]; // [W,R,G,B] [64 max levels]x
Log.BoxGammaCly [256]; // [W,R,G,B] [64 max levels]y
Log.BoxGammaCCT [256]; // [W,R,G,B] [64 max levels]CCT
Log.BoxGammaRes [24]; // [W,R,G,B] [Box(val,cc,mini,maxi),
deltauv(max,avg)]

Log.Halation [25]; // [Lum,%] [12], % halation
Log.Loading [25]; // [Lum,%] [12], % loading
Log.BxChk [7]; // [w,w,b,b,C,rows,cols] checkerboard

Log.Traverse [28]; // [neg,pos] [Lc,L1-
8,Cbavg,Cbmin,Cbmax,Ceiaj,ff]

Log.UnSUW [78]; //306-1:3 [1-
9,Avg,Min,Max,Non] [CCT,Lw,Lb,Cv,x,y] ([Non,x]=max du'v')
Log.UnSUC [225]; //306-4:5 [4] [[1-
9,Avg,Min,Max,Non] [L,CCT,x,y]+active,r,g,b]+Lw
Log.UnNUA [60]; //306-6a
10 [r,g,b,NU1,NUdu'v',5spares]+[10] [xpos,ypos,L,x,y]
Log.UnNUM [256]; //306-6b
10 [r,g,b,row,cols,5spares]+[82] [L,x,y]

Log.VaFPCR [36]; //306-1
[U,D,R,L] [angle,Lw,xw,yx,CCT,Lb,xb,yb,Cr]
Log.VaFPCO [40]; //306-1 [U,D,R,L] [angle+[r,g,b] [L,x,y]]

Log.RTmeas [8]; //305-1 Ton,Toff measured values,
sample rate, filter width (all ms),%fs,top/bottom cursor%,readings
Log.RTpat [6]; //305-1 Ton,Toff pattern
xorg,yorg,xsize,ysize,color,blinks/sec

Log.FLmeas [6]; //305-4 flicker nitems, time,
DClum(=-uncal), max flicker lev,maxflickerfreq,npoints
Log.FLfft [32]; //305-4 flicker fft ac coefficients
[0:nitems-1]
Log.FLdb [32]; //305-4 flicker db levels [0:nitems-
1]

```

```

Log.LineWidth[11];
//lwp0,lwp1,lwp2,hlw0,hlw1,hlw2,vlw0,vlw1,vlw2,hpklum,vpklum
Log.CvgTst[6]; //rgx,rgy,bgx,bgy,brx,bry
Log.Mtf[10]; //hlpeaks,hlpmtf,hlapvmtf,hlamtf,hlamtfstd
//vlpeaks,vlpmtf,vlapvmtf,vlamtf,vlamtfstd
Log.Dmdstats[10]; //hrast_avg_pk,hrast_avg_vly,hdmod,
//vrast_avg_pk,vrast_avg_vly,vdmod
Log.FFT[10]; //
Log.Tva[13]; //[max/avg] [vert/horz] [jit/swim/drift],RA
Log.Linstats[10]; //0 Pat.Columns,1 Pat.Rows,2 Pat.Vfreq,3 Pat.Hfreq
//4 Lin.samples,5 Lin.ysamples,
//6 Hlin,7 Vlin,8 HlinAlt,9 VlinAlt
Log.LinTiming[20]; // Name of timing file used
Log.LinDlthz[50]; // horz. delta values
Log.LinDltvt[50]; // vert. delta values
Log.Linh[50];
Log.Linv[50];

Log.Unilum[100]; //lum0,lum1,...lum 10x10 max
Log.Unixcol[100]; //ccx0,...,ccx
Log.Uniycol[100]; //ccy0,ccy1,...
Log.Unistats[10]; //rows,cols,pat.color,colorspace,
Log.Unixunits[10]; //string for x,u,or u'
Log.Uniyunits[10]; //string for y,v,v'
Log.Uniphot[20]; //string photopic units, fl,cd/m2,..

Log.Geostats[12]; //0 rows,1 cols,2 #items,3 xsize,4 ysize,
//5 height(mm),6 width(mm),7 Tilt(deg),
//8 maxerror(mm),9 avgerror(mm)
Log.Pinstats[10]; //0 tedge,1 redge,2 bedge,3 ledge
//4 tmaxerr,5 rmaxerr,6 bmaxerr,7 lmaxerr

Fpm.nTests=37; //number of tests
Fpm.selectedtests[Fpm.nTests]; //Tests selected to run this
time
Fpm.completedtests[Fpm.nTests]; //Tests completed to date
Fpm.selectedfns[5]; //Functions to run
Fpm.selectedtimings[MAXTIMINGS]; //Tests selected to run this time

```