Agilent 81200 Data Generator/Analyzer Platform Changes and Bugfix History

Date	81200/81210 Revision
21.02.1998	1.00
26.02.1998	1.01
05.03.1998	1.02
11.03.1998	1.03
12.07.1998	1.10
20.07.1998	1.11
23.07.1998	1.12
12.04.1999	2.00
27.09.1999	2.1
28.10.1999	2.11
13.01.2000	2.11 SP1
16.08.2000	3.0
17.08.2000	3.01
06.09.2000	3.02
12.10.2000	3.03
10.01.2001	3.04
31.01.2001	3.05
19.02.2001	3.06
29.03.2001	3.1
09.04.2001	3.11
03.05.2001	3.11 SP1
20.07.2001	3.50
25.07.2001	3.51
30.07.2001	3.52
22.08.2001	3.52_SP1
26.10.2001	3.52_SP2

Major Changes from Rev. 3.1x to 3.5x

Known Problems

- SCPI comment :DVT:INST:HAND:MMEM: is wrong, please do use instead:DVT:MMEM:
- Waveform-Viewer: wrong marker range. E.g. range 0..798ns marker vernier 0..798, markers slider 0..799
- Waveform-Viewer: markers sometimes not visible. E.g. range 0..798ns, window 745..753ns, 1ns/div., marker position A+B 746ns invisible.
- Waveform-Viewer: marker placement. Marker A set to 1 ns, Marker B set to 0 ns using the vernier. A to B = 0 displayed, both Markers are positioned at the same place.
- Waveform-Viewer: Transition on last Pattern from 0 to 1, sometimes this transitions is visible, sometimes it remains 0 depending on time slider position.
- Waveform-Viewer / Segment Editor interoperation: Waveform-Viewer keeps segments open. This results in the strange effect that if you make changes to a segment in the segment editor and want to cancel these changes by closing the segment editor and saying no to "Do you want to save your changes?" the changes actually remain in the buffer until the waveform-viewer is closed. Reopening the segment editor on that segment before closing the waveform-viewer still shows all the changes. Note that the changes are actually not on disk, just in a buffer so when all viewers of that segment are closed the changes are actually thrown away as desired.
- Waveform-Viewer display refresh problem: When moving marker A around sometimes the hex-values of bus-signals are redraw n at other locations
- Waveform-Viewer: after port delete sometimes forgets its signals. Example: have two ports, waveform-viewer open. delete first port -> the signals in the waveform-viewer vanish... Delete second port -> just the signals of the no longer available port vanish.
- Waveform-Viewer: empty waveform viewer with segment resolution 16 and 2ns/div
- External Reference Mode does not allow Data frequencies below the external reference clock frequency. E.g. for external reference 10MHz it is not possible to have user frequency below 10MHz. Same is true for 5, 2, 1MHz
- File Browser - Details: Multiple instead of single selection
- Sequence expressions that are syntactically incorrect might crash the GUI. Can only happen in combination with an incorrect remote program.

Graphical User Interface or Firmware Server doesn't respond

In case the Graphical User Interface or the Firmware Server doesn't respond in a system with multiple GUI's and Firewire connection you should upgrade to the newest I/O Library for Instrument Control. At least Revision J.02.00.01 or later is recommended. You can download the latest released version of the Agilent IO Libraries from http://www.agilent.com/find/iolib

GPIB - Gateway problem

The way the GPIB card is operated as non-controller (to provide the interface of an instrument) has been improved. Unfortunately, if a system is opened for access for the first time it can happen that an errorneous answer is transmitted via GPIB bus and the communication is blocked afterwards. A workaround for this behavior was opening all systems to be used in GUI sessions - which sometimes is not applicable in production environments. To FIX this behavior, the user should do the following:

For opening a system handle, always use the OpenHandleEx() function code which handles the errorneous answer and the timeouts properly.
(C/C++ sample follows)

```
// sample code for safely opening a system handle via VISA
ViStatus OpenHandleEx(ViSession vi, char *SystemName, char *HandleTemplate, char *Answer)
{
        ViStatus status;
```

```
        status= viQueryf( vi, ":DVT:INST:HAND:CRE? %s,\"DSR\",\"%s\"\n", "%t",
        HandleTemplate, SystemName, &Answer );
        if( status < VI_SUCCESS )
        {
                int i = 90;
                // in a loop, we try to read the answer for the request, if we had no
                // success in the viQueryf call (i.e. it timed out)
                while ((i > 0) && (status < VI_SUCCESS))
        {
        Sleep(50); // a little delay time where we don't touch the Interface
        status=  viScanf(vi,"%t", &Answer);
        i--;
        }
        if( status >= VI_SUCCESS )
        {
                // Now we have at least a result for the request. But due to the possible
                // interface reset, it can be a nonsens string. Because of that we do a
                // plausibility check here to get sure we have the right handle.
                 // If we suppose we didn't receive the right answer, we try to get it by
                 // issuing a useless request.
                // normally, the Firmware prepends an underscore to the template name
                // upon success, so SYSTA becomes _SYSTA or _SYSTAA
                if( strncmp(&Answer[1], HandleTemplate, strlen(HandleTemplate)) != 0 )
        {
        // handle did not contain the Template - must be wrong, we have
        // to do a re-read attempt, for safety we ask for the handle again
        status=  viQueryf(vi, ":DVT:INST:HAND:CRE? %s,\"DSR\",\"%s\"\n", "%t",
        HandleTemplate, SystemName, &Answer);
        while ((i > 0) && (status < VI_SUCCESS))
                {
                Sleep(50); // a little delay time where we don't touch the IF lines
                status =  viScanf(vi,"%t", &Answer);
                i--;
        }
        }
        }
        }
        return status;
}

// Main program
        #define HDL_LEN         64

ViSession Ag81250;              // Virtual instrument handle (Agilent 81250)
ViSession DRM;                  // Default Resource Manager
ViStatus status;                // Return status
char hdl [HDL_LEN]= {0};

        // This function returns a session to the Default Resource Manager
        // resource.
        status= viOpenDefaultRM(&DRM);
        ...
        // This function opens a session to the Agilent 81250 at GPIB address 11.
        // It returns a session identifier that can be used to call any other
        // functions to Agilent 81250.
        status= viOpen(DRM, "GPIB0::11::INSTR", VI_NULL, VI_NULL, &Ag81250);
        ...
        // Open a connection to the "DSRA" 81250 system
        status= OpenHandleEx(Ag81250, "DSRA", "SYSTA", hdl);
        ...
```

```
// Now if the request returned VI_SUCCESS, hdl contains the handle
// and can be used to program the 81250 system
status= viPrintf(Ag81250, ":%s:SGEN:GLOB:PER  2.5e-9\n", hdl);
...
```

Bug fixes:

Rev. 3.52 SP 2

- Trigger is wrong in external start mode (E4861A)
- Second 16 bits are doubled on external start (E4861A)

Rev. 3.52 SP 1

Module Selftest Errors might be shown at wrong Module (Problem introduced in 3.50 Release)

- some small memory leaks in firmware server when closing handles (Problem introduced with 3.00 Release)
- Sweep Delay does not reflect Frequency changes

Rev. 3.52

- P&P systemSelect reports error when using a vi that was already associated with a system (now correctly switches system)
- P&P segment type change fails when having multiple segments open
- GPIB daemon locked when creating a handle on a system with no running GUI
- The GPIB demon doesn't start on a Windows 2000 system.It's not only a problem of Win2000. The registry entry names for the GPIB card changes again. SystemController -> SystCtrl and BusAddress -> BusAddr. (Back to the roots)
- Standard Mode Sequence Editor PRBS/PRWS field sometimes not correctly updated.
- 62A/64A Property Handler too course, in some situations BIOS error messages may occur
- 63A/65A unnecessary Threshold Range Check in differential mode
- E4805A Clock Multiplier does not work > 666MHz (now correct error message)
- E4835A Frontend Mode not in Setting Export
- System locks up when stopping immediately after start during syncronization
- E4862A/E4864A possible error after setting import regarding parameter "width"
- Importing Segments as expression strings > 1024 Bytes length hangs the firmware
- E4838A in E4832A: Transition Times not changeable in Run Mode
- Two GUI handle conflict (_DSRA_OFF and _DSRA)
- Settig export via hp81200_settingExportToBuffer() reports error code but not error message
- Empty Front-End (within E4832A/E4841A timing parameter are not exported
- Module selftest not working when having master/slave system
- E4805B (Selftest error) VCO

Rev. 3.11 SP 1

- Netmeeting crash after ParBERT installation
- E4805B Selftest fails after measuring external frequency
- E4861A Fast Eye Mask fails on lower channel
- E4861A: timeouts too tight when measuring many bits
- E4861A & E4832A start/stop slower

Major Changes from Rev. 3.0 to 3.1

Bug fixes:

Rev. 3.01

- E4805B CPLD fixes: sporadic frequency counter selftest errors, sporadic sequencer memory counter selftest errors
- E4832A / E4861A Synchronization on a memory based PRBS broken
- P&P could only handle 25 connections to firmware
- GPIB daemon shows assertion after fresh startup and clicking on Settings (Workaround: press ignore)
- Torx Screw driver now shipped
- Highest Frequency (e.g. 666.667 MHz @ Segmentresolution 16) does not work. (Workaround 666.666MHz)
- E4805B holds ECL trigger lines High on Run
- numerous problems when installation directory uses more than 50 characters (no problem with default path)
- connect to server dialog endless loop when having specified an invalid IP adress.
- Parameter Editor incorectly causes implicit stop/start on close. This happens when focus was on a widget that causes stop/start
- FPGA consistency check error message after a setting load. This happens after a Deskew was done on E4832A Modules.
- Firmware server crash when using terminal index 0 for fast pass/fail queries.
- E4805B Selftest errors are shown as errors instead as reports in the selftest window
- GPIB daemon prevents VXI access on 2-Slot Controllers
- GPIB deamon does not store its settings into registry
- Config Tool now also enables GUI startup in Controlled predefined setting

Rev. 3.06

- E4805B sporadic frequency counter selftest error

Rev. 3.05

- Version Number in _HANDLE:IDN?
- No beeping during CPLD Hardware reprogramming
- P&P import into LabVIEW 6i documentation
- E4805B external Start not working for Segment Resolution >= 8
- Firmware Server crash during a client disconnect while the GUI queries for changes
- E4862A / E4864A output levels wrong in differential termination mode
- E4861A external start level sometimes high for a certain time before the first valid data bit comes out
- E4805B +24V Selftest too strict

Rev. 3.04

- E4805B Clock Input Sensitivity (was 800mV, now 200mV)
- run state query could sometimes return wrong state at E4805B

Rev. 3.03

- E4862A 2.6666667GHz Problem (BER > 0): CPLD change and BIOS drivers, HW change

Rev. 3.02

- GPIB daemon not working on E9850A Slot Controller

Rev. 3.01

- With NI based E9850A Controllers sometimes an I/O error "unsupported bus width" occured. As a workaround we now try several times before we give up

Major Changes from Rev. 1.0 to 2.11SP1

Note:
We urgently recomment all customers to upgrade
the current SW rev. to the latest SW rev. 3.52 SP2 for fee.

Please contact your local Agilent Representative to get the latest version on CD.