APSPA

Arbitrary
Pulsed
Semiconductor
Parameter
Analyzer

User's Manual

CNERF Collaborative Nonlinear Electronics Research Facility

APSPA Manual V 2.6.1 July 3, 1998

This document describes a sophisticated instrument designed for pulsed amperage-voltage measurement of two-port devices. It is called an "Arbitrary Pulsed Semiconductor Parameter Analyzer" (APSPA).

The information presented in this document is prepared in good faith. However, no responsibility can be accepted for correctness of the documentation, the APSPA system software, and the APSPA measurement system.

Anthony Parker, James Rathmell, and Jonathan Scott
Collaborative Nonlinear Electronics Research Facility
Department of Electronics
Macquarie University
Sydney, AUSTRALIA, 2109
email:tonyp@mpce.mq.edu.au
http://www.mpce.mq.edu.au/elec/cnerf/
phone: + 61 2 9850 9067
fax: + 61 2 9850 9128

Contents

Manual Additions

Since 2.5.1

- Section 7.4 has been added to detail software changes. Corresponding minor changes are reflected in the appropriate manual entries.

1 Getting Started

This chapter introduces pulsed-bias testing of two-port devices and provides an overview of the Arbitrary Pulsed Semiconductor Parameter Analyzer.

1.1 Quick-start Guide

1. For an overview of the Arbitrary Pulsed Semiconductor Parameter Analyzer (APSPA), refer to this chapter. A block diagram of the instrument is given in Figure 1.2 on page 1-5.

2. An overview of the instrument software is given in Chapter 2 . A quick-start guide at the beginning of Chapter 2 points to summary information on making basic measurements.

3. More detailed information on calibration and use of the instrument, and a description of its operation, is covered in Chapters 3 and 4 .

4. Assembly of the instrument is detailed in Chapter 5 . A quick-start guide at the beginning of Chapter 5 points to summary information on the assembly and set up of the hardware.

5. Specifications and trouble shooting information are given in the Appendix (Chapter 7).

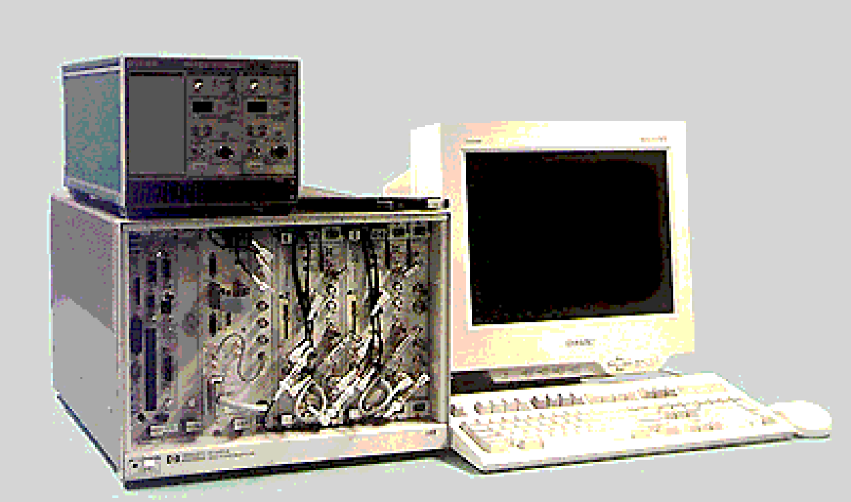

Figure 1.1: Photograph of the APSPA system.

1.2 Pulsed-bias Characterization

Conventional steady-state transistor measurements do not provide adequate information on the behavior of the device at high frequencies. The device's quiescent conditions, including temperature, do not vary during normal operation as they do during a 'step and sweep' steady-state measurement of characteristics. Moreover, rate dependent anomalies, which are prevalent in microwave devices, are not properly observed in the steady state.

Pulsed-bias measurement of a transistor gives characteristics at a specified quiescent condition. These are representative of the high frequency operation of the device. Characteristics well outside the safe operating area of the device, including breakdown, can also be measured.

A pulse measurement is performed by maintaining, for most of the time, a constant quiescent bias on the device under test. Periodically, a rapid *pulse* is applied to the device's terminals and during the brief interval of the pulse, the terminal voltage and amperage are measured. The pulses are kept short enough that the device's quiescent state is not disturbed, the device is not stressed, and rate dependent anomalies are avoided. The period between pulses is kept long enough to permit full recovery to quiescent conditions between pulses.

The Arbitrary Pulsed Semiconductor Parameter Analyzer (APSPA) is a sophisticated pulsed-bias measurement tool. It has been designed for flexible and rapid measurements ranging from routine production line measurement of device characteristics, to detailed analysis of transient and rate dependent anomalies. The system can also interface with a pulsed microwave network analyzer.

The APSPA system is fast and easy to use. In some cases it may even become the preferred instrument for quick measurement of steady-state characteristics.

1.3 Overview of APSPA

In basic usage the APSPA system is similar to a dc semiconductor analyzer. The latter performs steady-state measurements, while the APSPA performs measurements using high-speed pulsed conditions for *isodynamic* characterization of rate anomalies and for characterization of operation outside the safe-operating-area of the device. The term *isodynamic* refers to steady conditions of both thermal and electron trapping effects.

The APSPA system is comprised of a core set of VXI instruments, with some auxiliary equipment, and software for driving the instrument and measurements. The VXI instruments are installed, as described in Chapter 5 , in a single mainframe, as shown in Figure 1.1 . The APSPA instrument software, introduced in Chapter 2 , drives the instruments and orchestrates measurements.

1.3.1 Description of Hardware

Figure 1.2 shows the block diagram of the APSPA system. Figure 5.1 on page 5-5 shows the physical disposition of instruments in the VXI mainframe.

The basic pulsed-bias measurement system consists of the VXI modules and the current probes. There are four sub-systems:

- Two voltage-source amperage-monitor units, each comprised of
  - an HP E1340A arbitrary function generator to provide a sequence of pulse values,
  - one (or more wired in parallel) HP E1446A summing amplifiers to output the voltage pulses and add a dc bias, and
  - two channels of an HP E1429B 12-bit digitizer, one for recording voltage and the other for recording current. The latter is sensed via a Tektronix AM5030 amplifier and A6032, dc-50MHz, Hall-effect probe;
- One hardware timing controller — an HP E1450A 160MHz Timing Module — and a custom cable harness for distribution of the control signals. The control signals clock pulses from the arbitrary function generators, trigger voltage and amperage sampling, and provide trigger and RF power signals for an external pulsed network analyzer (if used); and

- One controller, an embedded PC, which also controls external instruments via a GPIB interface.

1.3.2 Description of Operation

In the basic mode of operation APSPA provides a stimulus to the device under test, which is a sequence of pulses added to a quiescent bias. The digital-to-analog converters within the summing amplifiers are set to deliver the desired quiescent bias (see Fig. 1.2).

Pulse Sequences

The arbitrary function generators provide pulses that are added to the bias and delivered to the device under test. The controller loads these values into the generators' memories in alternate locations, interspersed with zeroes (which produce bias values).

Test sequences are normally generated from start, stop and step values for Port 1 and Port 2 voltages. They may also be read from a data file, permitting arbitrary trajectories in the (Port 1, Port 2) voltage plane.

The pulse sequence can be randomly reordered to eliminate systematic anomalies in the measurement. This is the preferred mode of operation because an inadequate choice of pulse duration and periodicity shows up as 'noise' rather than a systematic shift in characteristics. More detail is given in Chapter 4 .

Measurement Procedure

To make a pulsed-bias measurement, the amplifier outputs are enabled, so that the test stimulus is applied to the device under test. The digitizers wait for the start of a measurement sequence, indicated by marker from the Port 2 arbitrary function generator, and then sample voltage and amperage values whenever triggered by the timing module. After the requested data has been measured, the stimulus signals are removed.

The controller sorts the sampled data, performs necessary averaging and scaling, and outputs the results.

Timing of Measurements

A pulse is generated whenever the timing module clocks the arbitrary function generators to a pulse value. A subsequent clock from the timing module returns the system to the quiescent state. This is repeated to produce a sequence of pulses. Independent clocks are provided for Port 1 and Port 2, and for an RF source of a pulsed network analyzer (if used). The relative timing of these clocks is user controlled.

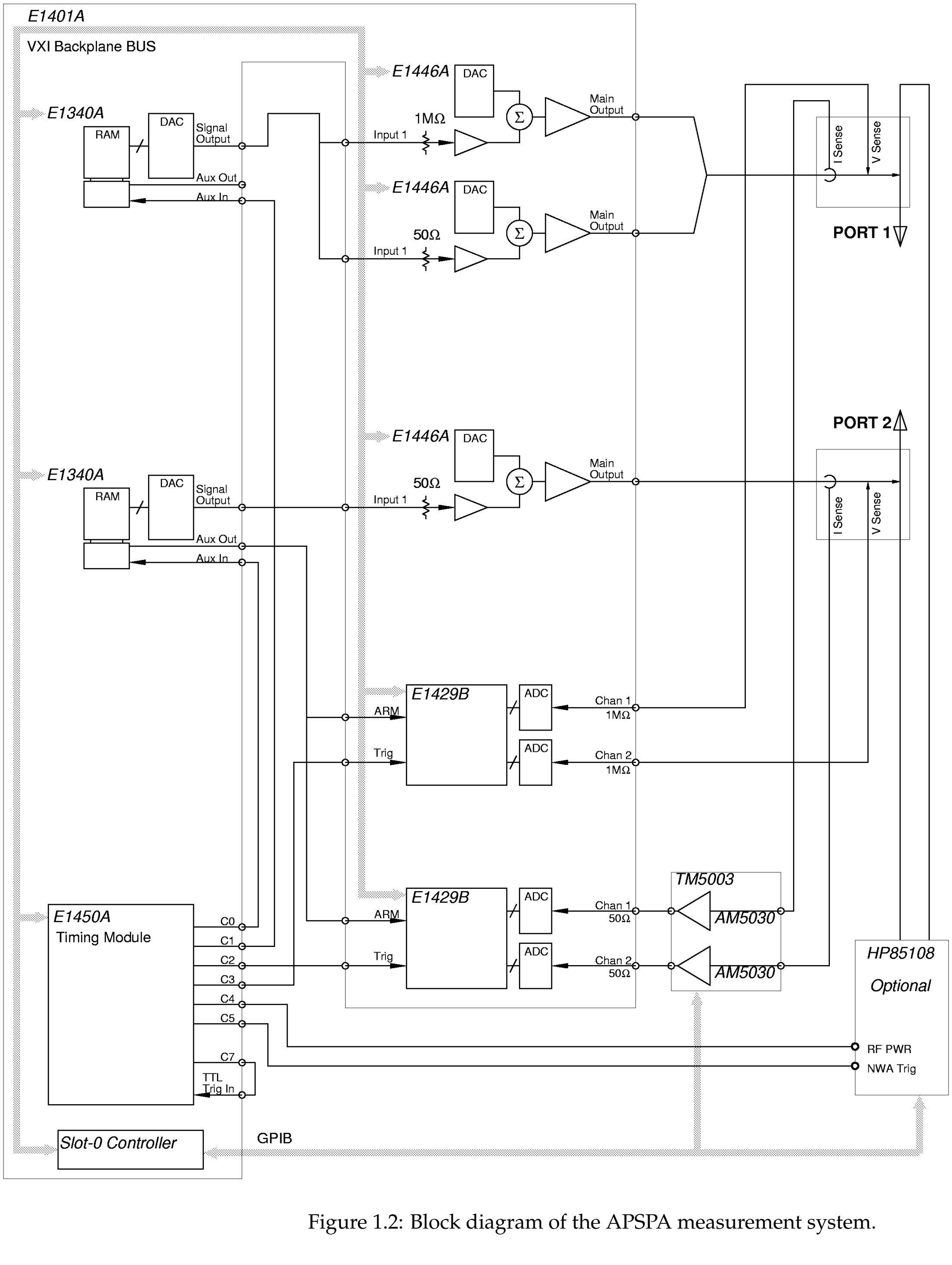

Figure 1.2: Block diagram of the APSPA measurement system.

The timing module also triggers the digitizers to sample voltage and amperage within the pulses. The time interval within the pulse during which data is considered 'valid' may be arbitrarily set. The digitizers can take more than one sample in a single pulse, and will continue through as many repetitions of the pulse sequence as is required to obtain a requested averaging factor.

Pulsed-RF measurements

If S-parameters are to be measured using an HP 85108A/L Pulsed RF Network Analyzer, then the RF power and trigger signals provided by the timing module are used.

Pulse values are measured one at a time, so that the network analyzer can acquire data for each frequency and for the number of repetitions required for averaging. The controller reads RF data from the analyzer for each pulse point requested.

Next Step

The first step, if not already done, in setting up an APSPA system is assembly of the instrument as described in Chapter 5 . Then the instrument software should be installed as described in Chapter 2 .

An overview of the APSPA software and its use is given in Chapter 2 . For more detailed information on use of the software and its operation, refer to Chapters 3 and 4 .

◇

2 Software

This chapter describes the APSPA instrument software. Information found here includes an overview of the software and a description of its installation and use.

2.1 Quick-start Guide

1. Consult Section 2.2 for information on setting up the operating system and installing the APSPA software. Consult Chapter 5 if the hardware has not been assembled.

2. An overview of the instrument software and the files it uses is given in Section 2.3

3. The operation, under software control, of the APSPA system during a basic measurement is described in Section 2.4

4. A tutorial guide to measurement of basic device characteristics is given in Section 2.5 . Pulse-domain, time-domain and S-parameter measurements are covered.

5. More detailed information on use and operation of the instrument is given in Chapters 3 and 4 , which give a software reference and cover advanced topics.

2.2 Installation

The APSPA instrument software is a command-line program, written in C, for controlling the pulse system. It is installed by simply copying the executable onto the system after preparation of the hardware drivers for the VXI and GPIB interfaces.

2.2.1 System Preparation

The instrument software requires a few Megabytes of RAM and must operate under Microsoft Windows 95, Windows NT, or a UNIX operating system. The complexity of a measurement sequence is likely to be limited only by available memory.

The hardware interface is through a Virtual Instrument Software Architecture (VISA) library (or formally a Standard Instrument Control Library (SICL)), which is linked into the APSPA executable. These libraries interface to hardware drivers that must be installed through the controller's operating system.

Installation Instructions for Windows 95

These notes apply to a RADEPC7 embedded controller starting from a clean installation of Windows 95. This includes installing the display drivers through the display control panel. The video card resources were IRQ 7 and I/O base 300hex. Windows 95 does not have native drivers for the EXM-13A video module. These can be obtained from RadiSys at http://www.radisys.com/Library/EXM/EXM13A-win95/.

An EXM-10A network card was also included in the prototype system, so the National Semiconductor AT/LANTIC EtherNode 16-ATS driver set for NDIS ETHAT 16-bit (refer to the RADEPC7 hardware documentation) was added to the hardware configuration.

Note that in the prototype, the EXM BIOS were changed, as outlined in the RADEPC7 documentation, to match the EXM-22 HPIB card in slot 2. New settings were:

Slot	ID	OB1	OB2	
0	FF	00	00	not used
1	ED	01	01	EXM-13A video card
2	D9	FF	00	EXM-22 HPIB card
3	F5	05	00	EXM-10A network card
4	FF	00	00	not used
5	FF	00	00	not used

Instrument Drivers Install the EPConnect/VXI for Windows 95 1.00, Win95 System Framework, from the disks supplied with the controller. (The controller may have been shipped with this already installed.)

Select the

`Programs→HP_IO Libraries→I_O Config`

item from the `Start` menu item. Then from the *Available Interface Types*, select the “RadiSys EPC-7 VXI”, and “RadiSys EXM-22 HP-IB” drivers.

The configured interfaces have default SICL names:

- `hpib7` for the EXM-22 HPIB card, and
- `vxi` for the VXI mainframe;

and VISA names;

- `GPIB0` for the EXM-22 HPIB card, and
- `VXI0` for the VXI mainframe.

These names will need to be included in the Hardware Configuration File (to be installed next).

2.2.2 APSPA Instrument Software Installation

To install the APSPA instrument software, first prepare the system as described in the previous subsection, and then

1. copy the executable `apspa.exe` to a location that is in the operating system `PATH`, and
2. create a Hardware Configuration File, described in Section 5.6 , named `apspa.hcf` in the same location. An `apspa.hcf` file is also included with the software distribution, but may need to be updated to reflect the actual hardware configuration.

If the name of the executable is changed then ensure that the Hardware Configuration File is also renamed to have the same base name with suffix `.hcf` .

2.2.3 Auxiliary Software

The APSPA software requires a command-line prompt. In Windows, the MS-DOS command prompt is suitable, although JP Software’s 4DOS (or 4NT) (http://www.jpsoft.com/) will give a more useful shell.

A UNIX environment is very useful for manipulating APSPA files and data. Mortice Kern System’s (http://www.mks.com/) MKS Toolkit provides a good set of UNIX tools for Windows.

2.3 Software Overview

The APSPA instrument is operated by executing the instrument software on the controller. The software uses a command line prompt, so Windows 95 or NT will launch it within a command prompt window.

Measurement instructions, measured data, and other reports are all communicated through files, which have plain text format. This facilitates seamless operation with other applications or operating systems.

2.3.1 APSPA Files

In a typical measurement scenario, the user prepares a *set* file, and then runs APSPA by typing

`APSPA` *file*

at the command prompt. Status and progress messages are sent to the console and *log* file during the measurement. At the end of the measurement, the pulsed-bias data is written to a *piv* file.

All files associated with a measurement have the same base name but have a different suffix. For example, a measurement made with a *set* file named `pulse1.set`, will produce a *piv* file called `pulse1.piv` and a *log* file called `pulse1.log`.

- There are three possible input files:

  file.set is the main file that contains the measurement instructions. Refer to Section 3.2 for a full description.

  file.scr is an optional script file, used to automatically sequence several measurements.

  file.lst is read when it is desired to take data at user-specified pulse points instead of the grid defined in the *set* file.

- There are four possible output files:

  file.log is produced during each measurement to report interim progress and diagnostic information.

  file.piv is produced during a pulsed-bias measurement. It contains the voltage/amperage data, in text form. More detail is found in Section 3.4 .

  file.pul is produced if a time-domain pulse-profile measurement is made. Time-domain voltage/amperage data are stored in this file.

  file.spm is produced if pulsed-RF measurements are made. The S-parameters are stored in this file.

2.3.2 Command Line Invocation

There are two common command-line invocations of the instrument software, one which produces a default *set* file and one which executes a *set* or a *scr* file.

The invocation `apspa -h` directs the built-in default *set* file, similar to that shown in Figure 2.1 , to the standard output stream. No measurements are performed. Thus

```
apspa -h > test1.set
```

produces a default *set* file called `test1.set`, which can be changed with any text editor. In a typical scenario, a user would produce a *set* file using this invocation and then edit it to produce the desired measurement conditions. The invocation `apspa -H` will produce a *set* file with all parameters annotated with comments.

The invocation

```
apspa test1
```

executes a measurement using the *set* file with base name `test1`. Upon successful completion, the output files will have been written to the same location as the *set* file.

Other command-line options, described in Section 3.1 , can be used for more control of the instrument. There are options to override *set* file parameters, to select hardware configurations, and to invoke scripted operation.

The APSPA instrument software command-line synopsis is fully described in Section 3.1 .

2.4 Description of Software Operation

The APSPA system produces pulse stimuli at two measurement ports and measures amperage and voltage, and possibly network S-parameters during each pulse. The operation of the APSPA system is controlled and described from four points of view:

1. *Timing* of the duration and period of each pulse event, and the instants when measurements are made,

2. *Bias* conditions to be established between each pulse and prior to recording data,

3. *Pulse Values* to be measured during each job commissioned by the user, and

4. *Job Control* for specifying and automating measurements made by the instrument.

The measurement job is divided into *job cycles* that are measured individually. Each job cycle is a periodic repetition of a sequence of *pulse events*, which is applied for a user specified stabilizing time and continued while data is sampled. Often a measurement will consist of only one job cycle.

2.4.1 Timing Control

For every measurement job, the APSPA system produces a continuous, periodic, sequence of *pulse events*. All pulse events associated with a particular job have identical timing.

Every *pulse event* consists of

- a *quiescent* interval during which a constant bias voltage is applied to each port, and no measurements can be made,
- a *pre-pulse* interval during which the bias is still applied and measurements can be made,
- a *pulse* interval during which a pulse is applied to each port and measurements can be made, and
- a *post-pulse* interval during which the bias is re-applied and measurements can be made.

2.4.2 Bias Conditions

The bias voltage applied to each measurement port is specified in the *set* file. The bias is maintained during the *quiescent*, *pre-pulse*, and *post-pulse* periods.

The system will also apply bias conditions during the initial and final stages of powering on the stimuli.

The APSPA software will normally sample the bias voltage in the *pre-pulse* period of each *pulse event*. A time-domain pulse-profile measurement can sample both the *pre-pulse* and *post-pulse* periods.

2.4.3 Pulse Values

Pulse values to be measured during each job commissioned by the user are specified in the *set* or *lst* files. Pulse cycles, with the different pulse amplitudes, are first sequenced by the software. A pseudo-random ordering is applied by default. If the total number of values is larger than can be handled in one *job cycle*, then the task in divided into a number of sequences of values. The resulting *jobs* are then executed.

This division is selected to minimize the total measurement time. The E1340A arbitrary function generators can support up to 2048 pulse values in each job cycle (depending on the complexity of the measurement). However, if one port uses a high-power pulser, which can handle only one pulse value per job cycle, then all requested values for the other port are sequenced into the same job.

In each job, the sequence of pulse values is repeated continuously during the measurement. Thus the stimulus applied to the device under test is a steady-state periodic signal.

2.4.4 Job Control

The measurements performed are specified in the *set* file. Normally a few samples of voltage and amperage are taken during each pulse event. These are averaged and output to a *piv* file.

Other possible measurements, which are requested in the *set* file, are a pulse profile (in a log or linear time domain) or network S-parameters.

The averaging factor, random sequence, instrument settings, and calibration, are controlled through the *set* file.

An important setting is the time that the system will wait after applying the pulse stimulus before making a measurement. This allows the device under test to settle to a steady state while the periodic pulse sequence is applied.

2.5 Measurement Tutorial

2.5.1 Launching APSPA

It is best to first launch the MS-DOS command prompt, or similar, and then invoke the APSPA software. This will allow use of the command-line options. Double clicking the APSPA software will allow an invocation with no options, but will request a *set* file base name for the job.

The easiest way to start a measurement is to generate a default *set* file. Move to a suitable working directory and invoke the APSPA software with the `-h` option to generate its built-in default *set* file.

For example, typing the following at the command prompt

```
apspa -h > pulse1.set
```

will create a file called `pulse1.set` which will contain a list of parameters similar to Figure 2.1 . No measurements are performed.

Invoking APSPA without any command-line arguments generates a brief command-line help message followed by a request for the *set* file base name:

```
c:> apspa
Arbitrary Pulsed Semiconductor Parameter Analyser            (AEP, JBS & JGR)
Version 2.6.0 (May 10 1998 16:10:10)  (c) copyright by CNERF 1995, 1996, 1997
Synopsis:

 C:\BIN\APSPA.EXE [-h|H] [-qln] [-ci] [-s[Sfile]] [basename[.set]]

   basename is the base name of input and output files.
   -h Sends a default set file to stdout and exits.
   -H Same as ’-h’, but adds comments and more parameters.
   -s Invokes scripted operation.
      Default is to use script parameter in set file.
      -s      takes script lines from stdin.
      -sSfile takes script from file Sfile.
   -c Select configuration block in file C:\BIN\APSPA.HCF
      Default is configuration 0
      -c      switches to configuration 1
      -ci     switches to configuration i
   -q Invokes quiet mode (Minimal messages to user).
   -n Force a pause for network analyser callibration.
   -l Suppresses generation of log file.

Please enter setup file base-name (or return to quit):
c:>
```

```
#Arbitrary Pulsed Semiconductor Parameter Analyser            (AEP, JBS & JGR)
#Version 2.5.1 (Dec 15 1997 12:51:13)  (c) copyright by CNERF 1995, 1996, 1997
vdq       = 2.0    # default was 0.0
vdstart   = -0.1   # default was 0.0
vdstop    = 7.9    # default was 1.0
vdstep    = 0.1
vgq       = 0.0
vgstart   = -1.0
vgstop    = 0.4    # default was 0.0
vgstep    = 0.1
wait      = 5
pulsetime = 7e-7   # default was 1e-5
quiestime = 10e3   # default was 90e-6
tsample   = 2e-7   # default was 3.5e-7
tsinset   = 1e-7
qpoints   = 1
average   = 16
tdinset   = 0
tqpre     = 2e-7
trfinset  = 1e-7
trftrig   = 0
nwacal    = 0
spms      = 0
idfsd     = 0.10
igfsd     = 0.01
shuffle   = 1
profile   = 0
logprof   = 0
listfile  = 0
nocal     = 0
zoutg     = 0
zoutd     = 0
#--- ------------------------------------------------------------ ----
```

Figure 2.1: Template set file for APSPA generated by the `-h` option and subsequently modified as noted by the comments.

2.5.2 Pulsed-bias Characteristics

An example of a typical measurement procedure is described here.

The first step is to edit the default *set* file, which was done for this example to give that shown in Figure 2.1 . The original values are recorded by comments after the appropriate parameter. Characters from the octothorp (hatch, hash, or pound sign) # character to the end of the line are ignored by the APSPA software. This allows detailed notes and comments, such as the device details *et cetera*, to be added to the file.

For this example, Port 1 was connected to the drain of a microwave MESFET and Port 2 to its gate, which is the assumed normal configuration. The parameter names reflect this assumption.

A grid of (Port 1, Port 2) voltage points to be pulsed has been set up by changing parameters `vdstart`, `vdstop`, `vdstep` for Port 1, and `vgstart`, `vgstop`, `vgstep` for Port 2. (Note that the chosen voltages may not be safe for some devices.)

The bias point for the test was set to 2 V at the drain by changing parameter `vdq`, and zero at the gate by parameter `vgq`.

The APSPA system will wait at least 5 s, set by parameter `wait`, after applying the pulse sequence before taking measurements. This allows the device response to stabilize to its new bias condition.

The pulse has been set to 700 ns by changing parameter `pulsetime`, and the period between pulses, during which bias is applied, has been set to 10 ms by changing parameter `quiestime`.

Data is gathered during a 200 ns interval, set by changing parameter `tsample`, that ends 100 ns before the end of the Port 2 pulse, set by parameter `tinset`.

At least 16 samples, as set by parameter `average`, are gathered at each pulse point and averaged. The voltage grid is repeated until sufficient data is gathered.

The bias is measured once (set by `qpoints`) 100 ns prior to the Port 2 pulse (set by `tqpre`).

The other parameters were not altered, and some of them do not influence the standard pulse test. See Section 3.2 for a detailed description of all the *set* file parameters.

The pulse measurement is performed with the modified parameters by invoking the APSPA software with the base name of the *set* file as follows.

```
apspa pulse1
```

This measurement asks for 1215 pulse points with a long quiescent period, but should complete in less than two minutes during which progress is reported to the user. When completed, a *log* file and a *piv* file will have been created in the same location as the *set* file. The head of the *piv* file should look something like the following.

```
\timestamped by apspa at Fri Dec 19 16:16:41 1997
\Vgs from  -1.000e0 to   400.0e-3 by   100.0e-3
\Vds from  -100.0e-3 to   7.900e0 by   100.0e-3
\t_q= 10.000e-3, t_p=  700.0e-9
\Qpt: -519.911e-6  2.00004e0  20.293e-3  249.86e-6
\Averages/pt = 20 (some at 40; 16 requested)
\Random seed = 1
\1215 lines
\Vgs         Vds          Ids          Ig            Index
81 4
 -963.73031e-3 -16.465143e-3  283.27410e-6  271.91003e-6 1443 1456
 -966.31877e-3  82.496392e-3  249.04334e-6  359.60233e-6 1216 1420
 -965.81877e-3  175.72716e-3  255.58180e-6  336.91002e-6 727 769
 -965.93800e-3  275.15023e-3  273.65872e-6  265.75618e-6 931 945
 -963.95339e-3  373.65023e-3  275.58180e-6  296.14080e-6 36 748
 -965.33031e-3  465.68869e-3  297.50487e-6  366.52541e-6 789 1509
...
```

A detailed description of the *piv* file format is given in Section 3.4 .

Another test can be performed to gather a dc measurement of the device by lengthening the pulse time. For this example the *set* file was copied to create a new *set* file.

```
copy pulse1.set dctest.set
```

The new file was modified to increase the pulse time to 10 ms, and reduce the maximum drain voltage to 5 V, because the power dissipation in the device is higher with longer pulses. The sample interval was increased to 1 μs and the averaging factor was also increased to exploit the extra data gathered. The affected lines are the following.

```
..
vdstart   = 0.0    # -0.1   # default was 0.0
vdstop    = 4.9    # 8.0    # default was 1.0
..
pulsetime = 10e3   # 7e-7   # default was 1e-5
quiestime = 1e3    # 10e3   # default was 90e-6
tsample   = 1e-6   # 2e-7   # default was 3.5e-7
..
average   = 128    # 16
..
```

This measurement should also complete in less than two minutes. When completed, new *log* and *piv* files will have been created. The head of the *piv* file should look something like the following.

```
\timestamped by apspa at Fri Dec 19 16:33:01 1997
\Vgs from  -1.000e0 to   400.0e-3 by   100.0e-3
\Vds from   0.0  to   4.900e0 by   100.0e-3
\t_q=  1.000e-3, t_p= 10.000e-3
\Qpt: -195.688e-6  2.00165e0  20.653e-3  196.00e-6
\Averages/pt = 147 (some at 294; 128 requested)
\750 lines
\Vgs         Vds          Ids          Ig           Index
50 4
 -1.0020193e0  8.7996872e-3 -10.078442e-6  203.37261e-6 0 750
 -1.0027982e0  107.17553e-3 -14.500210e-6  281.77397e-6 1 751
 -1.0030924e0  209.78608e-3 -33.377760e-6  269.18893e-6 2 752
 -1.0020397e0  308.69764e-3 -49.534223e-6  315.78757e-6 3 753
 -1.0035737e0  408.40001e-3 -17.731503e-6  280.92363e-6 4 754
 -1.0031434e0  508.91021e-3  22.234483e-6  262.21614e-6 5 755
...
```

The data in the *piv* files is formatted as ordinary text and can be imported into most graphing packages. The results from these examples have been plotted in Fig. 2.2 .

As a further example, the bias point in the pulse measurement was changed to 5 V at Port 1 and –0.3V at Port 2. Figure 2.3 shows the results.

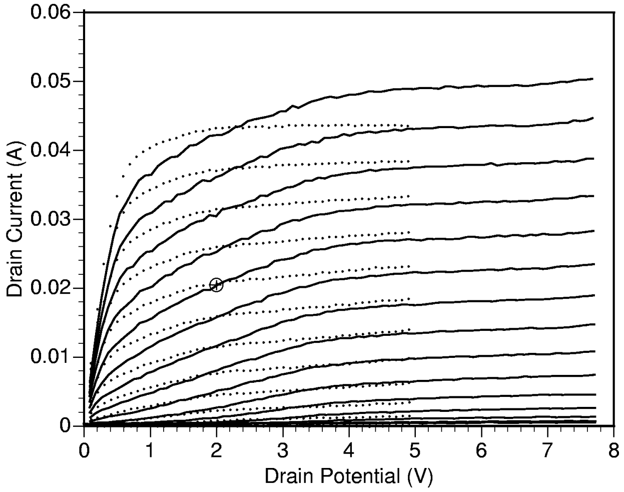

Figure 2.2: A graph overlaying the two pulse-bias characteristic examples described. The lines show the data for 700ns pulses with the bias point marked by the circled cross. The dots show the data for the 10ms pulses, which represent the dc characteristics.

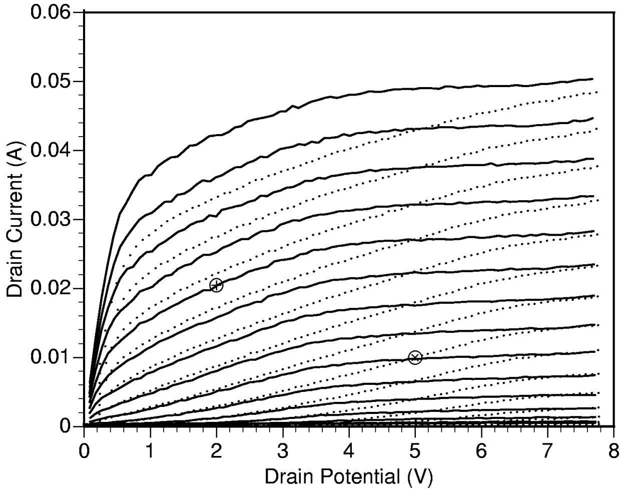

Figure 2.3: A graph overlaying the two 700ns pulse-bias characteristics at the two bias points indicated by the circled crosses. The dots show the data corresponding to a drain bias of 5 V.

◇

3 Software Reference

3.1 Command-line Synopsis

The APSPA command-line synopsis is as follows.

```
apspa [-h|H] [-qln] [-ci] [-s[Sfile]] [basename[.set]]
```

If no command-line arguments are supplied then a summary of this synopsis is printed to the screen. The brackets, `[]` indicate optional arguments and the vertical bar, `|` indicates alternative options.

basename is the base name of input and output files. If *basename* is not supplied, then a prompt will request it. The input and output files will be read from and written to the same location as the set file '*basename*`.set`'.

Unless other command-line options direct otherwise, APSPA will attempt to open the file '*basename*`.set`' and perform a measurement. If the *set* file is not found, then APSPA will attempt to open the file '*basename*`.scr`' as a script file and invoke scripted operation.

`-h` provides simple help by directing the built-in default setup file, similar to that shown in Figure 2.1 , to the standard output stream. Thus the command

```
apspa -h > file.set
```

will produce a *set* file called `file.set`, which can then be edited with any text editor. The *set* file parameters will be described in the next section.

No measurements are performed.

This method always yields a template appropriate to the current APSPA executable, and will therefore be more up to date than existing files or printed documentation.

`-H` is the same as `-h` but provides a full list of parameters with comments appended.

`-c` selects a configuration block in the Hardware Configuration File. The default configuration is 0, the first block in the HCF, which is also selected by `-c0`.

- `-c` switches to configuration 1, the second configuration block in the HCF.
- `-c`*i* switches to configuration *i*.

When a different hardware configuration exists, such as using a high-power pulser, the hardware is wired differently. Details of each alternate hardware setup are contained in the Hardware Configuration File, described in Section 5.6 .

`-s` Invokes scripted operation. The default is to use the script parameter in the *set* file, described in Section 3.2 .

- `-s` takes script lines from the standard input – the keyboard or redirected file.
- `-s`*Sfile* takes script lines from file *Sfile*.

`-q` Invokes quiet mode. Minimal messages are presented to the user. The copyright notice, termination statement and the title line of the configuration block are not displayed.

`-n` Forces a pause for network analyzer calibration. This causes a pause with the port outputs disabled but RF test set trigger and RF power gate signals active. The user can then calibrate the network analyzer with the correct timing signals.

`-l` Suppresses generation of log file.

3.2 SET File

The APSPA *set* file is the main file that contains the measurement instructions. It must have the following structure:

- The file's name must have the form *basename*`.set`. Its base name is be used to identify the file and to name other files used by the APSPA software.
- The *set* file must be prepared in plain text format.
- All characters from any of the symbols '#', ';', or '*', which indicate comments, to the end of the line are ignored by the APSPA software. This allows comments to be added after parameters. Also, any line starting with one of the comment symbols is ignored.
- Parameters are written as ordered pairs

  keyword = *value*

  where *keyword* is the name of one of parameters described below, and *value* is the numerical value to be assigned to that parameter. The = symbol is optional and is converted to a space before parsing.
- The end-of-line character has no significance accept to terminate a comment. Thus a line can contain many parameter assignments.
- The maximum length of a line must not exceed 255 characters.
- Parameter re-assignments will over-ride any previous assignment and a warning message will be issued.
- Parameters that are not included in the *set* file are set to their default value. Thus, an empty set file will use default values for all parameters. (These values will be reported in the *log* file.)

The following sections describe the parameters known to the APSPA software.

3.2.1 Voltage Parameters

The names for the voltage parameters reflect the assumption that Port 1 will normally be connected to the drain of a FET, and Port 2 will be connected to the gate. These can similarly be the collector and base, respectively, of a BJT.

vdq

vgq

The quiescent operating point is set with these parameters. The Port 1 quiescent potential is set to the value of `vdq`, and the Port 2 quiescent potential is set to the value of `vgq`.

The default values for both of these parameters are zero.

The APSPA software can generate a grid of (Port 1, Port 2) voltage points using six parameters. The parameters

vdstart

vdstop

vdstep

set the Port 1 values to be used to construct the grid. The default parameters give eleven voltages stepping by 0.1 V from 0.0 V to 1.0 V, inclusive.

The parameters

vgstart

vgstop

vgstep

set the Port 1 values to be used. The default parameters give eleven voltages stepping by 0.1 V from –0.1 V to 0.0 V, inclusive.

The grid will not include a potential that is outside the interval between the start and stop values. It will always include the start value but not necessarily the stop value, depending on the step value. Where there is an exact multiple of the step magnitude between the limits, both limits are included. Thus stepping from 0 to 1 by 0.1 gives eleven values (0, 0.1, 0.2, . . . , 1.0).

If the sign of the step parameters is opposite to the start-to-stop direction, then the grid is constructed in reverse and will end at the start value. Thus stepping from 0 to 1 by -0.3 gives values (0.9, 0.6, 0.3, 0.0).

A step value of zero will generate only one voltage at the start value.

The voltage grid is constructed by stepping the Port 2 (gate) values and sweeping the Port 1 (drain) values. This ordering will be randomized before the measurement unless parameter `shuffle` is set to zero.

The default values will generate 121 points.

The six voltage grid parameters can be ignored in favor of reading a file of voltage points by setting parameter `listfile`.

listfile

Setting `listfile` to any non-zero value directs the APSPA software to read a *lst* file containing a list of voltage points to be measured.

The start, stop and step parameters of the *set* file are ignored.

The list file must have a name of the form *basename*`.lst` and have the same base name as the *set* file. Each line in the file that starts with two numbers is interpreted as a voltage point. Text following these two numbers and other lines are ignored. The two numbers are, in order, the Port 2 (gate) and the Port 1 (drain) voltage values. Only one voltage point per line is permitted.

This is useful both for the profile mode (described below), and for producing arbitrary trajectories. Such arbitrary trajectories can be used to mimic operation with a resistive or reactive load line, especially with relatively short quiescent intervals.

The default setting for `listfile` is zero, which causes the generation of a voltage grid using the start, stop, and step parameters.

3.2.2 Timing Parameters

Several parameters set the timing of voltage changes and measurements within each pulse event. These are described in this subsection, and are shown pictorially in Figure 3.1 .

The length of each pulse event is set by the sum of the parameters `pulsetime` and `quiestime`. The sum of these two parameters is the pulsing period.

pulsetime

The parameter `pulsetime` sets the width of the Port 2 pulse. The default value is 10 μs. A value of less than 1 μs should be used with care and requires careful setting of the other timing and calibration parameters.

Note that `pulsetime` defines the width of the Port 2 pulse. The Port 1 pulse is affected by parameter `tdinset`, relative to `pulsetime`.

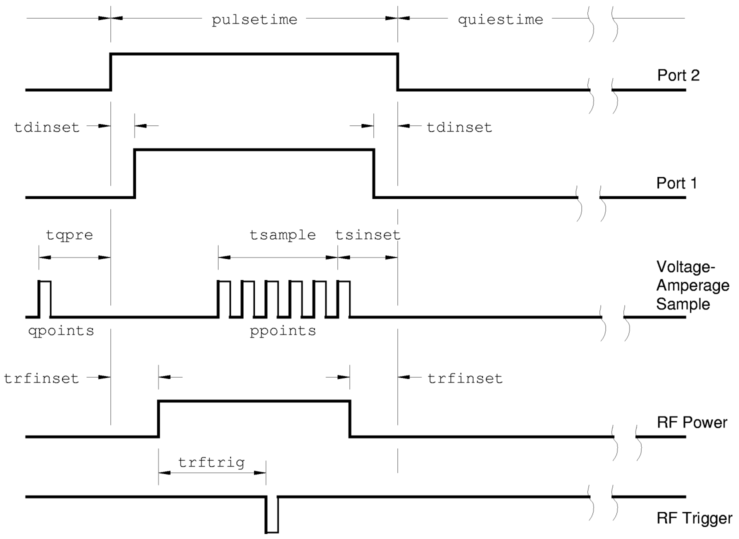

Figure 3.1: APSPA timing diagram showing the relationship between timing parameters.

quiestime

The parameter `quiestime` sets the interval between the Port 2 pulses. During the quiescent interval the bias potentials are applied to both ports. The default value is 90 μs.

Note that `quiestime` defines the interval between Port 2 pulses. The Port 1 quiescent interval is affected by parameter `tdinset`.

tdinset

The parameter `tdinset` sets the duration by which the Port 1 pulse is inset from the Port 2 pulse. Thus, if `tdinset` is set to 100 ns, the Port 1 pulse will start 100 ns after the Port 2 pulse, and finish 100 ns before the Port 2 pulse finishes, making the Port 1 pulse 200 ns shorter in total. A negative value will make the Port 1 pulse wider than the Port 2 pulse.

The default setting of `tdinset` is zero, so that the Port 1 and Port 2 pulses are coincident.

The times at which voltage and amperage samples are taken are set by parameters `tsample`, `tsinset`, `ppoints`, `tqpre` and `qpoints`.

tsample

The magnitude of the parameter `tsample` sets the length of the measurement interval during which samples are made. The sample triggers are spread evenly throughout the interval set by `tsample`. The maximum number of samples that can be taken in this interval is limited by the maximum sample rate (20 MHz for the HP E1429B) of the digitizers.

The default value of `tsample` is 350 ns, which gives 8 samples at 20 MHz.

It is desirable to set this parameter to make measurements at the latter part of each pulse where signals have settled.

The sign of this parameter has no significance.

tsinset

The time from the end of measurement interval to the end of the Port 2 pulse is set by `tsinset`. The default value is 100 ns.

If this parameter is reduced then care should be taken to ensure that the measurement interval does not extend past the end of either the Port 1 or Port 2 pulse. Cable length delays can affect this.

Parameter APSPA Manual V 2.6.1 July 3, 1998/tsinset/ can be set to a negative value, which is useful for measuring pulse profiles.

ppoints

The maximum number of samples taken in the measurement interval is set by the value of `ppoints`. No more than `ppoints` samples will be taken. However, fewer samples are taken if the maximum sample rate of the digitizer would otherwise be exceeded.

The default value for `ppoints` is 128. Thus the default number of samples in measurement intervals less than 6.35 μs will be limited by the 20MHz maximum sampling rate of the digitizers.

Setting `ppoints` to a low value will reduce the sample rate, so that the samples are spread evenly over the measurement interval `tsample`.

Setting `ppoints` to 1 will give one sample per pulse at the end of the measurement interval.

The bias condition is also sampled, during the pre-pulse period of the pulse event.

tqpre

The parameter `tqpre` sets the time of the bias sample relative to the start of the Port 2 pulse. This parameter should be set to prevent the Port 1 pulse interval from overlapping the bias sample. The pulse measurement interval (`tsample`) is truncated to prevent overlap with the bias sample.

The default value for `tqpre` is 200 ns. Thus setting `tdinset` to -200 ns will cause the beginning of the Port 2 pulse to be sampled, rather than the bias condition before the pulse.

qpoints

The parameter qpoints sets the number of bias samples taken in each pre-pulse period. These are always measured at the maximum sample rate of the digitizers (20 MHz). The parameter tqpre defines the position of the last of the qpoints samples.

The default value is 1, which is more than adequate because the average over all pulse events is returned. The pre-pulse intervals are all at the same bias point.

This parameter is ignored and no *bias* measurement is performed if profile is non-zero, which forces a time-domain measurement.

3.2.3 Job Control Parameters

Several parameters control the sequencing and type of measurement to be made.

wait

The parameter wait sets the settling time with stimulus before sampling begins. This time allows the device under test to stabilize at the average power and voltage levels of the measurement.

The default value for wait is 5 s.

average

The parameter average sets the minimum number of samples per data point that must be averaged to reduce noise.

If this averaging factor can be satisfied by the number of points taken in one tsample interval, then only a single job cycle is sampled. Otherwise, the sequence of pulse events will be repeated to complete the averaging. If more than average samples can be accommodated within the number of sequence repetitions, then these are used and a message to this effect is included in the header of the *piv* file. For more detail refer to Section 4.1 .

The default value for average is 16.

idfsd

The parameter idfsd sets the full scale range of the current probe and amperage digitizer.

If this parameter is set to a value that is too low, or there is excessive dc offset from the current probe, then overloads will be reported by the digitizer.

The default value for idfsd is 100 mA.

igfsd

The parameter `igfsd` sets the full scale range of the current probe and amperage digitizer.

If this parameter is set to a value that is too low, or there is excessive dc offset from the current probe, then overloads will be reported by the digitizer.

The default value for `igfsd` is 100 mA.

shuffle

The parameter `shuffle` sets the seed for the random number generator used to sequence pulse events in pseudo-random order.

If `shuffle` is set to zero, then the order is sequential. That is, in the order that the voltage grid was constructed or read from the *lst* file.

Because the value of `shuffle` is the seed for the random number generator, the same value will always generate the same point ordering. Different values will generate different sequences.

Data are always output in sequential order in the *piv*. However, one or two index numbers appear at the end of each data line. These numbers give the actual position of the pulse event in the job cycle's sequence of pulse events.

The default value for `shuffle` is 1, which will give pseudo-random ordering.

profile

When `profile` is non-zero a time-domain pulse profile measurement is performed.

At most `ppoints` samples are gathered in each pulse event over an interval set by `tsample` and `tsinset`. The pulse event is repeated `average` times and averaged samples are written to the *pul* file. The parameter `qpoints` is ignored and no bias measurement is performed. Refer to Section 4.2 for more detail.

The default value for `profile` is 0, which disables time-domain measurements.

logprof

The parameter `logprof` is ignored if `profile` is zero.

When `logprof` is non-zero a time-domain pulse profile measurement is performed with samples taken on logarithmic-spaced intervals (or as close as is practical with the hardware). Refer to Section 4.2 for more detail.

The pulse event is repeated `average` times and averaged samples are written to the *pul* file.

The default value for `logprof` is 0, which gives a linear time domain measurement (if `profile` is non-zero).

cal

The parameter `cal` sets the number of calibration iterations performed before each measurement. The calibration determines the gain and offsets of the digitizers and amplifiers. The accuracy of the calibration is improved with each iteration, although little improvement is gained after two iterations.

In each iteration, the digitizers are used to sample levels without application of a stimulus to determine dc offsets. Then a measurement is made with zero bias applied to determine amplifier offset errors, and finally a measurement is made with bias applied to determine gain errors. This latter gain measurement is omitted on ports that do not have zero output impedance, as set by `zoutg` or `zoutd`.

The default value for `cal` is 1, which provides a reasonable calibration.

Setting `cal` to zero omits the calibration.

Setting `cal` to –1 performs a zero offset calibration only. This provides a rapid calibration of the current probes, which tend to drift over time, without the overhead of a full calibration.

Setting `cal` to zero or –1 in the second line of script files can speed up the total measurement.

nocal

When `nocal` is non-zero the automatic calibration of dc offsets is omitted. Setting `nocal` to 1 forces `cal` to zero.

The default value for `nocal` is 0, which enables calibration.

zoutd

zoutg

The parameters `zoutd` and `zoutg` set the output impedance of the Port 1 and Port 2 amplifiers respectively.

The default for both is zero.

Values of 0 and 50 Ω are always supported by the HP E1446A amplifiers. Where multiple HP E1446A amplifiers are used in parallel, interim values are also possible; for instance, two amplifiers allow 25 Ω, and three amplifiers allow a value of about 17 Ω.

Where bias networks are used in the measurement setup, values of 0 Ω are appropriate. For direct connection, a tradeoff with device stability exists, and a value of 50 Ω, especially on the gate, is often necessary.

Note that the voltage measured by the digitizer will be that at the device, rather than the open-circuit voltage to which the amplifier is set. The difference will vary depending on the load and output impedance setting.

3.2.4 RF Measurement Parameters

The application of an RF signal and the measurement of S-parameters are set by the parameters `spms`, `nwacal`, `trfinset`, and `trftrig`.

spms

When `spms` is non-zero a network S-parameter measurement is performed. In this case, the RF signal will be applied during the pulse, and the resulting RF measurements will be read from the HP 85108A/L and stored in an *spm* file.

The default value for `spms` is 0, which disables S-parameter measurements.

Note that if `spms` or `nwacal` are non-zero, the RF gate is active and any pulse measurement will include the RF signal, which may affect the *piv* file data (unless the RF network analyzer is not connected).

nwacal

If `nwacal` is non-zero then APSPA will pause to permit calibration of the RF network analyzer. The user is prompted to either continue the measurement or exit APSPA. The Port 1 and Port 2 stimuli are disabled but the timing module *C4 - RF GATE* and *C5 - RF SAMPLE* signals are active.

This allows calibration with the same timing as the measurement.

The default value for `nwacal` is 0, which omits the calibration. A calibration pause can also be forced with the `-n` command-line option.

Note that if `spms` or `nwacal` are non-zero, the RF gate is active and any pulse measurement will include the RF signal, which may affect the *piv* file data (unless the RF network analyzer is not connected).

trfinset

The parameter `trfinset` sets the inset of the RF pulse relative to the Port 2 pulse. A negative setting will make the RF pulse wider than the Port 2 pulse.

The default value for `trfinset` is 100 ns.

trftrig

The parameter `trftrig` sets the position of the RF sample trigger relative to start of the RF pulse.

This trigger signal synchronizes the HP 85108A/L sampling within the RF pulse. Note that the sample will occur at a fixed delay time, set from the network analyzer's stimulus menu, after the APSPA trigger is received.

The APSPA trigger must occur at least 350 ns prior to the desired RF sample time. This is the minimum trigger delay that can be set in the HP 85108A/L.

The default value for `trftrig` is 0 ns.

3.2.5 Calibration Parameters

The APSPA software adds variable time displacements to each event to compensate for delays in the instruments and cables.

By default, the time displacements are read from the Hardware Configuration File and should not need altering. It is also possible to add further displacement by using *set* file parameters.

timeadj

If `timeadj` is zero (the default setting), then only the time displacements read from the Hardware Configuration File (refer to Section 5.6) are used.

Setting `timeadj` to 1 adds the values of the following parameters to those read from the Hardware Configuration File. This allows the user to add a different length of cable or external active components and yet retain the correct interrelationship of the control signals.

The user can also use the time displacement parameters to implement pulses with unequal insets.

tddrain

The value of `tddrain` sets the time displacement for Port 1.

tdgate

The value of `tdgate` sets the time displacement for Port 2.

tdsampi

The value of `tdsampi` sets the time displacement for the amperage digitizer sample.

tdsampv

The value of `tdsampv` sets the time displacement for the voltage digitizer sample.

tdnwa

The value of `tdnwa` sets the time displacement for the gating of RF power.

tdrf

The value of `tdrf` sets the time displacement for the RF sample trigger.

A negative displacement causes the event to occur sooner. The default for all displacements is zero. The resolution of the timing adjustment is 6.25 ns.

3.2.6 Miscellaneous Parameters

vdmax

vdmin

vgmax

vgmin

A software safety-check is included to prevent the user requesting measurements that exceed specified voltage limits.

The parameters `vdmax` and `vdmin` set the absolute voltage limits permitted at Port 1, and the parameters `vgmax` and `vgmin` set the absolute voltage limits permitted at Port 2.

The APSPA software will abort a run if any requested voltage is outside the ranges defined by these values. Typically they might be set to a device's maximum ratings. Note that the drain current is not known *a priori*, so no current limiting is implemented by the hardware.

For both ports, the default minimum and maximum limit voltages are –20 V and +20 V respectively. This effectively disables the safety check.

script

The APSPA software can operate in a mode whereby multiple runs are carried out with only minor changes from one to the next. For instance, 1000-point grids can be measured for different bias voltages simply by having a script file that resets `vdq` and `vgq` in sequential lines. The output data is written to the one output file. Refer to Section 3.3 for the format of script lines and the script file.

The parameter `script` sets the mode of scripted operation.

If `script` is set to 1, then the standard input (keyboard or redirected file) is read.

If `script` is set to 2, then the script is read from an *scr* file, which has the same base name as the *set* file.

The default setting is 0, which disables scripted operation. The `-s` command line option overrides any setting of parameter `script`.

3.2.7 Diagnostic Parameters

digits

The value of the `digits` parameter specifies the precision (number of decimal places) of output data.

This parameter can be used to control the size of output files, as these are all in text format.

The default value of 8 guarantees adequate resolution but a value as low as 4 may be sufficient.

leaveon

The `leaveon` parameter inhibits the switching off of stimuli at the end of a run.

This is intended for diagnostic use, especially in conjunction with timing adjustments, or proper preparation of monitoring instruments such as an oscilloscope.

The default value is zero, which shuts off the stimulus outputs at the end of a run.

3.3 Script File

Scripted operation of the APSPA software can be through lines in an *scr* file or lines entered from the standard input. In either case the format of the script is same. The `script` *set* file parameter and the `-s` command-line option set scripted operation.

The end of each line in a script file invokes a measurement job. That is, the end-of-line character, either at the end of the line in the file or generated by the return key, will start a measurement.

The exception is a line ending in the backslash character `\`, which indicates a continuation onto the next line.

The APSPA software starts by reading the *set* file and setting the set file parameters as directed. If scripted operation has been invoked, either by the `script` parameter or the `-s` option, then the first line of the script is read. A measurement is performed once an end of line character, which is not preceded by the backslash character `\`, is read.

If no *set* file if found, then APSPA attempts to read an *scr* file with scripted operation invoked. Parameters not assigned by the first script *line* will take on their default values.

Each script line contains one or more *set* file parameter assignments. The parameters are altered in accordance with the assignments and remembered thereafter unless re-assigned in a later script line.

The format of each script line is the same as the *set* file except that lines starting with a comment character must end with a backslash.

An empty line, or end of file, will terminate execution of the APSPA software. Thus blank lines are NOT permitted in a script file.

Example

The following lines will perform three measurements, each at a different Port 1 bias. A calibration is performed only once because the calibration settings are retained for the following measurements.

```
# These first 2 lines are comments ending with    \
#                                                 \
cal = 1         #  calibrate
cal = 0 vdq = 2  # change bias, ommit calibration
vdq = 3         # new bias, still ommit cal
# end
```

The output data is written to the one output file with the same base name as the *set* file. Other output files may be generated if the script invokes different types of measurements, such as of S-parameters or in the time-domain.

```
\timestamped by apspa at Tue Feb 11 13:51:50 1997
\Vgs from  -4.000e0 to   200.0e-3 by   200.0e-3
\Vds from   0.0  to   6.800e0 by   200.0e-3
\t_q=  2.000e-3, t_p=  500.0e-9
\Qpt: -1.94536e0  2.69016e-3 -292.03e-6  258.60e-9
\Averages/pt = 129 (some at 258; 128 requested)
\770 lines
\Vgs          Vds           Ids          Ig           Index
35 4
 -3.9000962e0  127.85839e-6  67.354535e-6   -2.0677761e-6 0 770
 -3.8991951e0  196.25188e-3  -134.19584e-6  -2.1452955e-6 1 771
 -3.8986718e0  389.74025e-3  -343.49816e-6  -1.6414196e-6 2 772
 -3.8990400e0  583.77125e-3  51.850660e-6   -2.2615746e-6 3 773
 -3.8983714e0  781.77512e-3  -262.10281e-6  -2.9592490e-6 4 774
 -3.8992435e0  978.84876e-3  -471.40513e-6  -2.8235901e-6 5 775
 -3.8989334e0  1.1769495e0   -76.056311e-6  1.9514971e-6  6 776
```

Figure 3.2: First lines of a typical *piv* file.

3.4 PIV File

The *piv* file contains the pulse response data, in plain text format. There is a defined structure, which allows automatic parsing of this file by other applications.

Lines starting with a backslash (\), are comments.

All *piv* files carry a header consisting of a number of comment lines. Figure 3.2 shows the first few lines of a typical *piv* file.

The first comment line is a time stamp. This is the only line whose position and content are guaranteed.

The next two comment lines indicate the grid as determined from the *set* file. In this example, the gate (Port 2) was stepped from –4 to 0.2 V in 0.2V steps, a total of 22 points. Likewise the drain (Port 1) limits produce 35 steps. The pulses were 0.5 μs and spaced apart by 2 ms.

The next comment line gives the bias voltage and amperage for both ports: V_{gs}, V_{ds}, I_g and I_{ds}. The naming reflects the assumption that a FET is connected to the system. This ordering of values (Port 2 voltage, Port 1 voltage, Port 1 amperage, Port 2 amperage) is used throughout the file, and is stated in a comment line just above the main data (along with the pulse event index indicating its position in the job cycle).

Next there is a comment line indicating the averaging that was applied to each measurement. Averaging of 129 has been applied as a minimum. (In fact three data were obtained in each pulse, from a `tsample` of 100 ns and a 20 MHz digitizing rate, and 43 pulse events were required to equal or exceed the 128 averaging requested, giving 129.)

The next comment line indicates the number of points in total (770, being 35 times 22). This number was rounded up to 1024 by replication of 254 of the required points, so 254 values will have been averaged 258 times. Refer to Section 4.1 . This is reflected in the comment line concerning averages.

The comment line identifying the columns of the data finishes with the field `Index`, and indeed the first data line is followed by the numbers 0 and 770. The second line is followed by 1 and 771, *et cetera*. This indicates `shuffle` was set to zero, so that the pulse events were not randomly reordered. A comment line stating that the data was measured in random order, and giving the value of `shuffle`, would be present if reordering had been specified.

The first four data columns give Port 2 and Port 1 voltage and amperage values, one point per line. Subsequent index columns show where each point was positioned in the job cycle sequence, so that it is possible to work out what points followed or preceded a given point. In this example, the 255th data line finishes with the single index number 254, as it is the first point that was measure only once per job cycle, and was thus averaged only 129, not 258 times.

Finally, the line with two integer-fields, 35 and 4, indicates how the measured data is organized. This line says that 35 lines follow with 4 measured fields. These points will correspond to one nominal gate (Port 2) voltage value. After the 35 lines, another two-integer-field line will herald the next block of 35 data lines.

3.5 PUL File

The *pul* file contains the time-domain pulse profile data, in plain text format. The format is very similar to the *piv* file. There is a header consisting of comment lines followed by the time, voltage and amperage data. There is one header and data block for each grid point (or *lst* file point) specified in the *set* file.

The main difference is that the first data column is time in seconds. The zero-time point is set to the beginning of the Port 2 pulse.

Note that requesting pulse profile on a normal characterization grid can produce very large amounts of data.

3.6 SPM File

The *spm* file contains the S-parameter data, in plain text format. A header of similar format to the *piv* file is followed by lines with nine fields, the first being frequency, and the next eight being the real and imaginary parts of the four S-parameters.

The file is divided into blocks corresponding to each bias point.

Note that a great deal of data, and a very long measurement sequence, can result if S-parameters are requested across a fine voltage grid.

◇

4 Advanced Topics

4.1 Details of Averaging

The HP E1340A arbitrary function generators fix the number of pulse points at 2048 (or 1024 when sampling close to or past the end of the Port 2 pulse). If fewer points are requested (or fewer can be taken in a single job cycle because of external hardware limits such as with an RF network analyzer or an external high power pulser) the number of pulse points is first rounded up to a sub-multiple of 2048 by the replication of some of the pulse values. This submultiple constitutes one job cycle and is replicated to fill up the 2048 points. For example, if 500 points are requested, 12 of them will be duplicated to make a job cycle of 512 points. This job cycle is replicated 4 times.

The job cycle is repeated until sufficient data is sampled to perform the requested averaging. If this can be satisfied with the number of points taken in each pulse (fixed by parameter `ppoints` or the digitizer speed and the available time) then only one job cycle is used.

The averages are calculated with all data that is sampled. In the example above, 12 points will be averaged twice as much as is requested because they are repeated in the job cycle. Further, if the averaging factor cannot be satisfied in n pulse visits, but is over satisfied in $n + 1$ visits, the higher number is used, and the resulting averaging factor will exceed the requested averaging factor, but this will not have taken any extra time.

Where more than 2048 points are requested, they are broken into job cycles of 2048 points or less, and processed separately as above. Note that shuffling, if requested, is applied before points are divided into job cycles. Also, the different job cycles might give different quiescent power dissipations.

4.2 Pulse Profiling Considerations

The time-domain pulse profile measurement is useful for looking at the extrema of current and voltage during the pulse, which allows determination of regions of minimal transients for sampling pulse data. Inclusion of the pulse rise and fall regions is, of course, highly undesirable when *not* measuring pulse profiles! Time-domain data is also used to extract the time constants associated with a device through post-processing of the data.

It is also the best mode to use when determining timing calibration delays.

The parameters `tsample` and `tsinset` set the time interval of the pulse profile measurement.

Note that a pulse profile is taken for *every* requested pulse measurement point. This can generate enormous amounts of data if a large grid is requested with many samples. It is best to restrict the grid by resetting the start, stop and/or step parameters, or by use of a listfile, so that manageable amounts of data are produced.

Logarithmic profiles can be very useful in displaying time constants quickly. Refer to D. N. Swingler, "A Differential Technique for the Fourier Transform Processing of Multicomponent Exponential Functions", *IEEE Transactions on Biomedical Engineering*, vol. BME-24, no. 4, July 1977, pp. 408–410. The log profile option dramatically reduces the size of data sets required when applying such algorithms.

However, when a logarithmic pulse profile is requested, the timer task becomes much more complicated. Although the timer algorithm will do its best to compensate, log-spaced sequences tend to require more resources. For this reason it may be impossible to carry out any reasonable approximation to a log-spaced sampling sequence for long cycle times. There will be a limit to the complexity of sequence that is possible.

However, small changes in the demands for a log profile, such as changing from 512 to 500 points, or a small change to the timing parameters, may allow successful compilation of the measurement timing sequence. For this reason, it may be helpful to "fiddle" the input parameters if a problem is encountered.

4.3 Operation

In the basic mode of operation, the arbitrary function generators provide pulses to be delivered to the device under test. The digital to analog converters within the amplifiers are loaded with a value that produces the desired gate and drain quiescent points.

Scaling is done on each run to set the amplifier gain and arbitrary function generator output amplitude to ensure that the best possible dynamic range is obtained for the particular voltages to be delivered. Also, a calibration test is run to allow the cancellation of amplifier offsets via software modification of the data.

The amplifier outputs are enabled (the Port 2 output first, and then Port 1 shortly afterwards, to ensure that the drain bias is not present without control of the gate), and then the test stimulus is applied to the device under test.

The digitizers are programmed to arm on the Port 2 arbitrary function generator's end-of-sequence marker, and to digitize voltage and amperage values when triggered by the timing module. After the data has been measured, the stimulus signals are stopped and the bias voltages are removed, Port 1 first.

Pulsed-bias characteristics

Test data are normally generated from start, stop and step values for Port 2 and Port 1 voltages. They may also be read from a data file, permitting arbitrary trajectories in the (Port 1, Port 2) voltage plane.

The digitizers will take as many samples as required in a single pulse, and will repeat the job sequence of pulses as often as is required to obtain a requested averaging factor. The time interval within the pulse during which data is sampled may be arbitrarily set. As data is read from the digitizers, it is averaged and stored. If more data is required than will fit in the digitizer's half-megabyte memory, the measurement is repeated and more data is read.

Pulsed-RF measurements

If S-parameters are to be measured using an HP 85108A/L pulsed RF network analyzer, then there will be only one pulse point per job cycle. This is because the network analyzer requires the same pulse to appear for each frequency and repeated enough times to complete its averaging. There will be a job cycle for each pulse point.

4.4 Calibration

The subsection describes a procedure for adjusting the timing calibration coefficients in the hardware configuration file.

4.4.1 Pulse Timing Calibration

The most important aspect of APSPA operation is timing of the pulse and measurement events. The following describes a procedure for calibrating the timing of events.

The port drives, voltage and amperage sense should be connected to the bias network that will be used for device measurements. Use equal length cables to connect the two device ports and the timing generator 'Marker Out' to an oscilloscope. Use the timing module's Marker Out to trigger the oscilloscope.

The aim is to set the timing of events at the device, which is the oscilloscope in this setup. The timing coefficients are used to adjust the relative position of each event.

Run the following set file:

```
vdq     = 0
vdstart   = 1     vdstop    = 1     vdstep    = 0
vgq     = 0
vgstart   = 1     vgstop    = 1     vgstep    = 0
wait    = 0
pulsetime = 400e-9 quiestime = 1600e-09
tsample   = 400e-9 tsinset = 1e-7  tqpre=0
idfsd     = 0.02   igfsd     = 0.02
cal       = 0
leaveon   = 1
profile   = 1
timeadj  = -1
```

This should leave a pulse event running that can be comfortably viewed at 100 ns per division.

The timing parameters are relative quantities, so one may be left at zero while the others are adjusted. The default values in the Hardware Configuration File supplied with the APSPA software were selected to account for the delay from the timing module to the device under test.

The reference event is the Port 2 pulse, and the beginning of this pulse is considered time zero. To set this, use the oscilloscope to observe the rising edge of the Timing Module's C0 - Port 2 Clock, and the Port 2 drive at the device. Set the hardware configuration file parameters to the negative of this value – in particular `tdgate`. The Port 1 time displacement `tddrain` is likely to be the same, unless a different hardware configuration is used for this port.

It is recommended that the start of the edge, rather than a midpoint, be used because the various events will have different rise or fall times. (When the user specifies a sample at the end of a pulse, this should be a point on the flat-top, not half way through a transition.)

4.4.2 NWA Timing Calibration

This subsection describes the setting of time calibration for the NWA signals.

Starting from the factory preset state, set the HP 85018 to wide band operation with the menu

```
system→more→pulse config→Detector Wide BW
```

and set the pulse width to 1 μs so that the pulse profile can be seen over a shorter window.

From the `Stimulus Menu` set `more→trigger mode→External` and `Trigger Delay` to 400 ns

Set Stimulus start and stop to the same low value (2 GHz for the HP 85110A test set), so that the RF can be observed with an oscilloscope.

To observe the RF signal on the bias pulse, connect the APSPA Port to a force input of a bias network and the RF test set to the RF input. Connect an oscilloscope to the 'bias + RF' output of the bias network. Trigger the oscilloscope from 'Marker Out' of the APSPA timing generator. Select an appropriate S-parameter, so that the port being observed is driven by the test set.

It is also possible to select `Domain→pulse profile` to observe the RF pulse. Do this, then select from the `Parameter Menu`, `a1` or `a2` to observe the driven port of the RF test set. A useful display can be established by selecting `stimulus menu→number of points→51`, reducing `Stimulus stop` to 5.3 μs, and setting `Format Menu→linear magnitude`, with the `Response Scale` set to a suitable value.

In both cases, the following set file should generate a suitable timing sequence:

```
vdq = 0.0
vdstart = 0.2   vdstop = 0.2     vdstep    = 0.0
vgq = 0.0
vgstart = -0.2  vgstop = -0.2    vgstep    = 0.0
wait      = 0
pulsetime = 2000e-09 quiestime = 8000e-09
trfinset  = 5e-07    trftrig   = -1000e-9
nwacal    = 1     spms      = 1 cal     = 0
```

From the `Stimulus Menu` set `more`→`trigger`→`Trigger Delay` to 1400 ns to account for the value of `trftrig` in the set file. The extra 1 μs is added to shift the RF burst into an interval that is visible when the network analyzer is in pulse profile mode.

Running this set file will cause APSPA to pause with the timing signals running, so that the network analyzer can be calibrated.

Within the bias pulse, the RF should start 500 ns after the beginning of the edge of the bias pulse and end 500 ns prior to beginning of the end edge of the bias. The timing parameter, `tdnwa`, should be adjusted to correct the position of the RF burst.

In the pulse profile display, the RF pulse should be evident and begin to rise at 1 μs and begin to fall at 2 μs. A corresponding RF burst should be visible on the oscilloscope. The timing parameter, `tdrf`, should be adjusted to shift the RF trigger, so that the RF burst occurs in this interval.

◇

5 Hardware Assembly

This chapter describes the hardware required for an APSPA system. Information found in this chapter includes a list of hardware required, a description of hardware assembly, and the hardware configuration file that describes the setup.

5.1 Quick-start Guide

The basic steps in assembling the APSPA system are:

1. Acquire the required hardware. The notes in Section 5.2 provide a guide for selecting hardware options. A functional system can be assembled with the core items listed in Table 5.1 .
2. Assemble the VXI mainframe. Recommended module settings and mainframe slots are summarized in Table 5.2 . Section 5.3 describes how to adjust the setting of each module.
3. Install front panel cables as described in Section 5.4 .
4. Assemble the external hardware as outlined in Section 5.5 .
5. Check the Hardware Configuration File. Section 5.6 describes the Hardware Configuration File and Table 5.4 on page 5 -19 defines the format of a configuration block within the file.
6. Go to Chapter 2 for information on installation of the APSPA software.

5.2 Hardware Manifest

The core set of equipment required for an Arbitrary Pulsed Semiconductor Parameter Analyzer (APSPA) is listed in Table 5.1 . (The items listed are those used in the prototype systems.) Alternatives, notably the mainframe or embedded controller, would be suitable, but have not been tested by CNERF.

The main components of the APSPA system are assembled in a VXI mainframe. Current probes are external to the mainframe. The system will also work with an RF test set and high-power pulsers.

The following notes will aid in selection of suitable hardware:

- Any thirteen-slot VXI mainframe that can deliver the following minimum power, air flow and current capacities is suitable (examples are an HP E1410B or a Tektronix VX1410).

	I_{peak}	$I_{dynamic}$
+5 V	23.4 A	1.4 A
–5 V	18.6 A	0.5 A
–2 V	8.6 A	0.4 A
+24 V	2.0 A	1.5 A
–24 V	1.9 A	1.5 A
+12 V	2.7 A	0.7 A
–12 V	0.8 A	0.4 A
Power	260 W	
Air flow	3.8 l/s	0.8 mmH_2O

- An embedded controller is essential for timely transfer of measured data. An MXI interface to an external computer can be considered. See Section 5.3.1 on installation of the embedded controller, on page 5-6.

- Although a current-sense resistor is an option supported by the APSPA system, an external dc-50MHz current probe is recommended. The current-sense resistor option is susceptible to common mode signal injection.

- The Tektronix AM5030 programmable amplifier with a GPIB interface is recommended for the dc-50MHz current probes. The AM503 non-programmable amplifier is also supported, but this will require manual adjustment of the current range settings.

- Two HP E1403B carriers are required for the HP E1340A generators.

- Additional paralleled amplifiers are required for a bias current between 200 mA to 600 mA.

- The APSPA system provides trigger and RF control signals for an HP 85108A/L pulsed microwave network analyzer. A GPIB interface to the network analyzer is required.
- External high-power pulsers are supported if currents up to 10 A are required.

Table 5.1: APSPA Core Equipment Manifest.

Qty	Part No.	Description
1	HP E1401B	C-size, VXI high-power mainframe, 13-slot
1	HP E6233A	VXI Pentium PC Controller - WinNT or Win95 equivalent add HPIB and Ethernet support add memory if more than 32Mb required
1		Computer Monitor
1		Keyboard to suit the PC Controller
1		Mouse to suit the PC Controller
1	HP E1450A	160MHz timing module (This is a component of HP's D20 digital test system)
2	HP E1340A	Arbitrary function generators, B-size
2	HP E1403A	A/B to C-size P1 active adapters for the E1340As
2	HP E1446A	Summing amplifiers / DAC
3	HP E1446A	Optional amplifiers for higher currents
2	HP E1429B	Two-channel 20MSa/s, 12-bit digitizers
2	TEK A6302	Tektronix dc-50MHz current probes
2	TEK AM5030	Tektronix GPIB programmable current probe amplifiers
1	TEK TM5003	Tektronix mainframe for AM5030s
1	HP 10833B	2m GPIB cable
4		6in BNC-BNC cables
2		12in BNC-BNC cables
1		BNC tee adaptor
6		BNC right angle adaptors
4		36in BNC cables to connect to the DUT
0-6		6in BNC-BNC cables (2 per extra summing amplifier)
0-6		BNC tee adaptors (2 per extra summing amplifier)
1		CNERF patch cable
1	APSPA Software	Measurement interface and hardware driver software

5.3 Assembly of VXI Modules

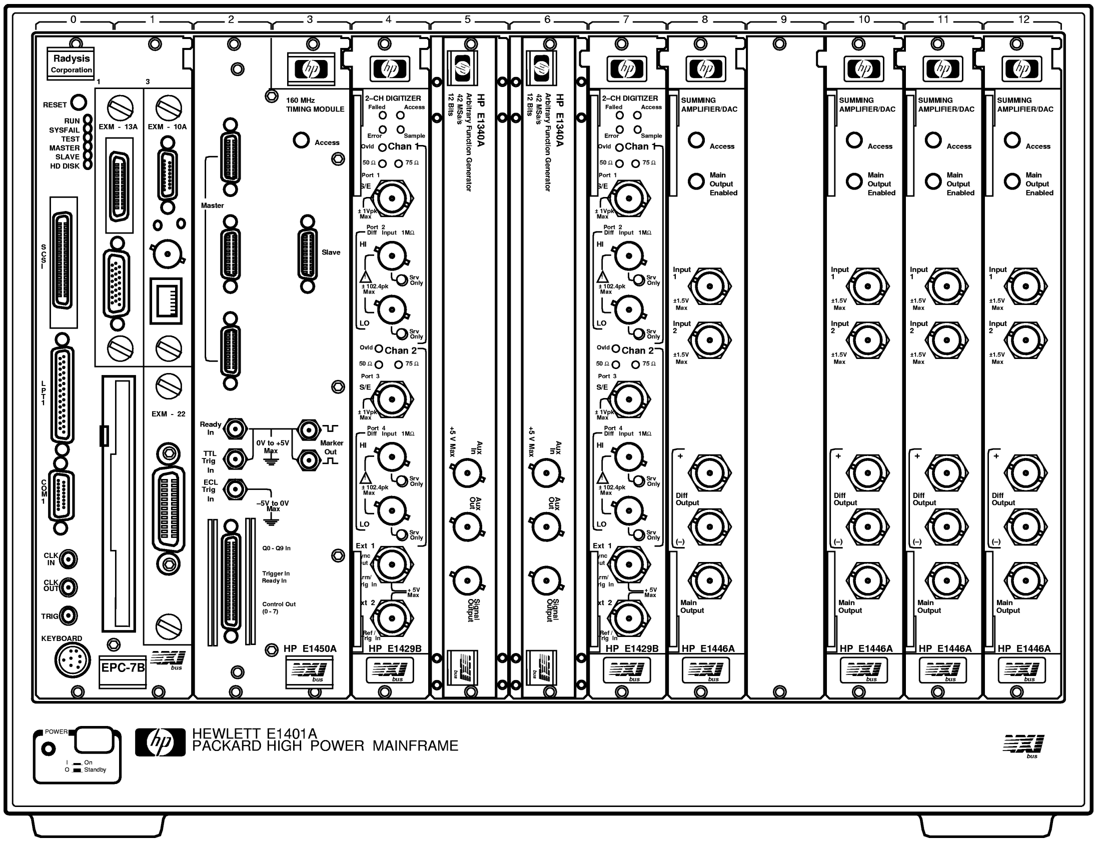

Figure 5.1: A typical APSPA configuration assembled into an HP 1410A high-power mainframe.

The main components of the APSPA system are assembled into a VXI mainframe.

Each module populating the VXI mainframe must be assigned a unique logical address (ULA) and be prepared for installation as described in this section. Record the ULA assigned to each module, as each ULA must be entered in the Hardware Configuration File described in Section 5.6 on page 5-16.

A typical system in an HP E1401A high-power mainframe is shown in Figure 5.1 . This system uses three HP E1446A amplifiers for Port 1 (drain), which can be connected in parallel. The mechanical position of the modules, other than the controller, is not critical, but the layout of this figure is recommended, so that inter-module cables can be arranged neatly.

A summary of the module settings used in the prototype system is given in Table 5.2 . Instructions for setting each module are given in the following subsections.

5.3.1 Embedded Controller

The slot-0 position in the VXI mainframe has back-plane connections for the controller of the mainframe. Thus the embedded controller must occupy slot 0.

A RadiSys EPC7 embedded VXI 486DX PC has been used in the prototype. Other embedded controllers, such as Pentium and UNIX based processors, or an MXI interface should be able to operate the APSPA software. Porting to these platforms can be arranged on request.

A GPIB slot-0 controller is not recommended because it cannot sustain a reasonable data transfer rate with the digitizers.

Before installing the controller, check that it has been configured as a slot-0 controller. Jumpers on the rear of a EPC7A, shown in Figure 5.2 , set this configuration.

5.3.2 Timing Module

The timing module generates control signals for the APSPA system. The HP E1450A, which is part of the HP Model D20 Digital Functional Test System, provides the necessary programming ability.

Before installation, check the timing module's ULA. The factory set logical address, ULA = 136, as shown in Figure 5.3 , is suitable.

The timing module must have its MASTER/SLAVE Jumper fitted between the front panel Master and Slave D-connectors. Also, a 50Ω load must be installed on the *ECL Trig In* SMB connector. Both the jumper and the load should have been shipped with the module.

Table 5.2: Settings for VXI mainframe modules. (An asterisk indicates factory settings that should not need altering)

Slot	Module	ULA	Other	Fig. Ref.
0	EPC7A	*	*Set to slot-0 Controller	Fig. 5.2
1	EPC7A			
2	E1450A	*136	Install the Master/Slave Jumper	Fig. 5.3
3	E1450A		and a 50Ω load to ECL Trig In	
4	E1429B(I)	41	*Bus Priority BG3	Fig. 5.4
5	E1340A(2)	73	Fit in a B-size carrier	Fig. 5.5
6	E1340A(1)	*72	Fit in a B-size carrier	Fig. 5.5
7	E1429B(V)	*40	*Bus Priority BG3	Fig. 5.4
8	E1446A(2)	87		Fig. 5.6
9				
10	E1446A(1a)	*88		Fig. 5.6
11	E1446A(1b)	89		Fig. 5.6
12	E1446A(1c)	90		Fig. 5.6

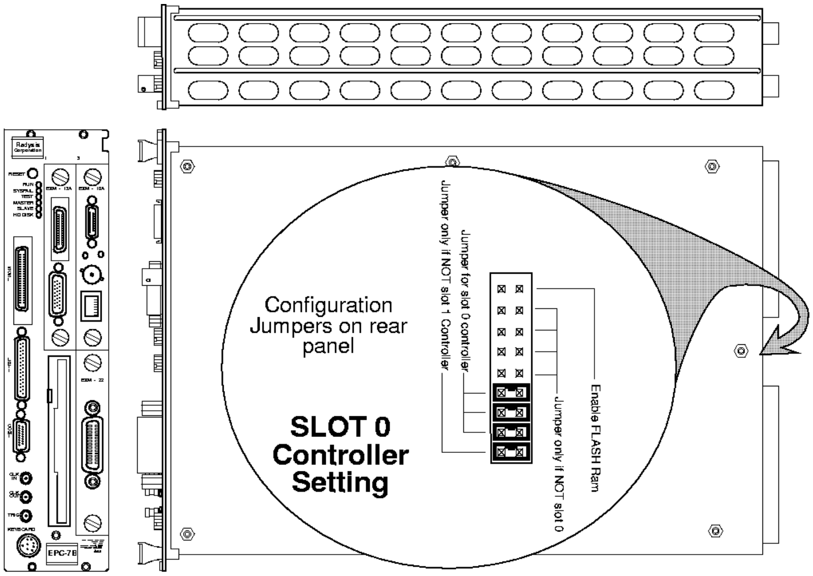
Radisys Corporation
RESET
RUN
SYSFAIL
TEST
MASTER
SLAVE
HD DISK
KEYBOARD
EPC-7E
Configuration Jumpers on rear panel
Jumper for slot 0 controller
Jumper only if NOT slot 1 Controller
Enable FLASH Ram
Jumper only if NOT slot 0
SLOT 0 Controller Setting

Figure 5.2: Configuration of a RadiSys EPC7A as a slot-0 controller.

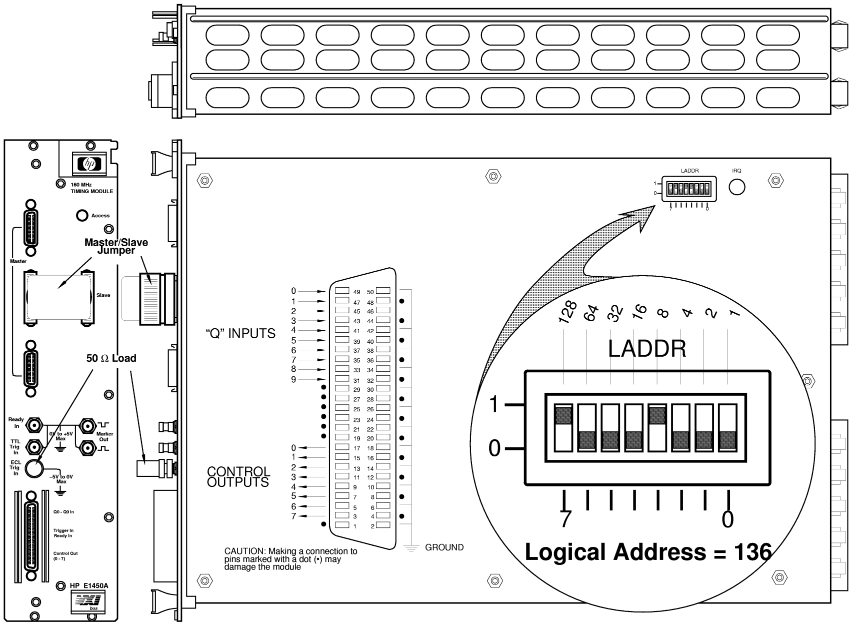

Figure 5.3: Use the switches on the side of the HP E1450A to set its unique logical address. The factory setting is suitable. Note that the Master/Slave Jumper is fitted and a 50Ω load is used to terminate the unused ECL Trig In connector.

5.3.3 Digitizers

A total of four digitizer channels is required to measure voltage and amperage at two measurement ports. The HP E1429B dual-channel digitizers are recommended because they provide 12-bit resolution at a reasonable sampling rate (20 MSa/s). The HP E1429A, which does not support VXI local bus transfers, has not been tested in an APSPA system but should work equally well because APSPA does not use the local bus.

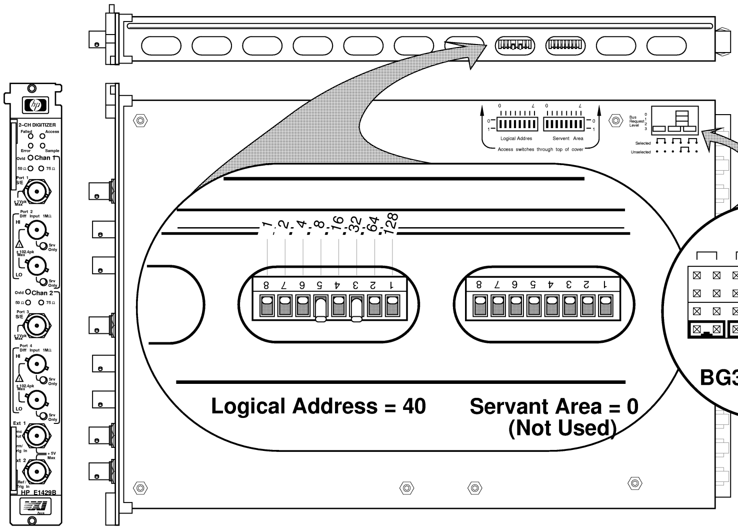

Figure 5.4: Use the switches on top of the HP E1429B to set the module's unique logical address. The bus request level should be set to BG3.

Before installation, each digitizer's ULA must be set. One, say for voltage measurements, may be left at the factory set logical address, ULA = 40, as shown in Figure 5.4 . The other, for current measurements, must be given a unique address, say ULA = 41.

The HP E1429B also has an adjustable bus request priority, which is the priority at which the module can request use of the VXI Data Transfer Bus. This should be left at the factory setting of BG3 – the highest priority.

5.3.4 Arbitrary Function Generators

The arbitrary function generators provide pulses that are applied to the device-under-test (DUT) via amplifiers. The HP E1340A function generators provide capacity for up to 2048 pulses with 12-bit resolution. These are size-B modules, so they must be fitted into VXI Size-B to Size-C adapters, such as the HP E1403B carrier.

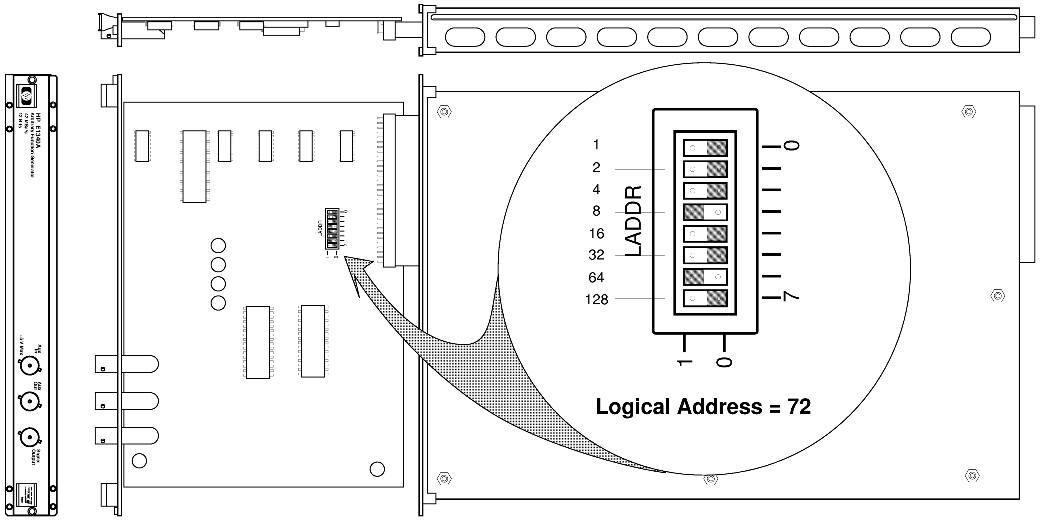

Figure 5.5: Before fitting an HP E1340A into its B-size carrier, set its logical address with the switches on the circuit board of the module.

Before installation, each generator's ULA must be set. One, say for Port 1, may be left at the factory-set logical address, ULA = 72, as shown in Figure 5.5 . The other, for Port 2, must be given a unique address, say ULA = 73.

5.3.5 Summing Amplifiers

The summing amplifiers provide dc bias and feed test pulses to the DUT. The HP E1446A amplifiers include a 16-bit digital-to-analog converter, which is used to set the dc bias. A variable gain amplifier adds the pulses provided by arbitrary function generators. Amplifiers can be paralleled to increase the current capability of the measurement system.

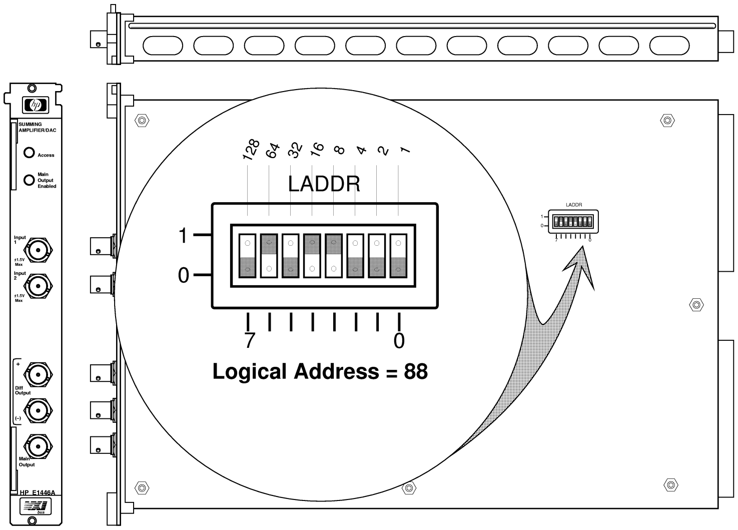

Figure 5.6: Before fitting an HP E1446A into its B-size carrier, set its logical address with the switches on the circuit board of the module.

Before installation, each amplifier's ULA must be set. One, say for Port 1, may be left at the factory-set address, ULA = 88, as shown in Figure 5.6 . Any other amplifiers that will be wired in parallel with the first Port-1 amplifier should be given consecutive addresses, say 89, 90, 91,. . . . The amplifier for Port 2 should also be given a unique address, say ULA = 87.

If a port is equipped with more than one amplifier, then the amplifier with the lowest ULA must be used as the base ULA in all configurations defined in the Hardware Configuration File. The input impedance of the other amplifiers is set high, so that they will not load the system when not used.

5.4 Front-panel Wiring

Front panel wiring is primarily via a Patch Cable and some BNC Cables. Figure 5.7 shows the cable layout used in the prototype.

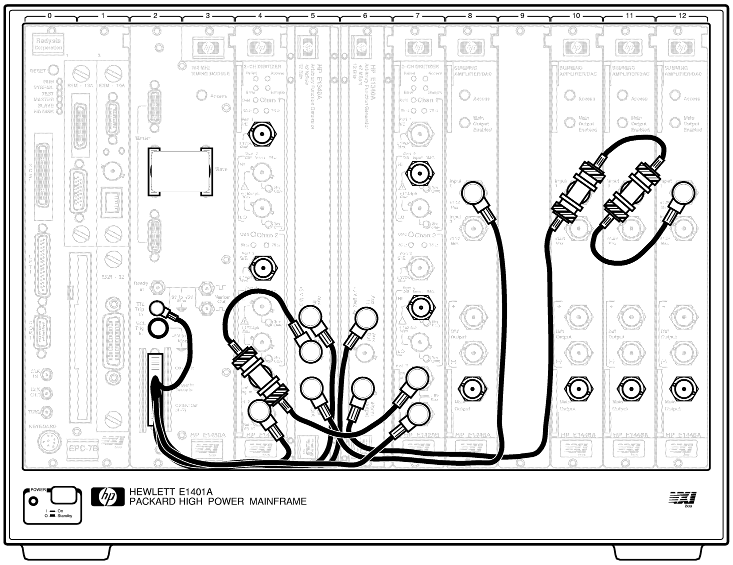

Figure 5.7: Diagram of front-panel wiring. Table 5.3 lists the interconnections required.

Note that timing module requires the MASTER/SLAVE Jumper and a 50Ω termination for the *ECL Trig In* SMB connector (see Figure 5.3).

Patch Cable

The patch cable has a 0.050″ Mini-D connector, which plugs into the timing module, and *flying* leads, which connect to the appropriate modules.

The CNERF cable is designed to handle the high-speed trigger signals required by the digitizers.

Note that the timing module can suffer catastrophic damage if an incorrect connection is made to its Mini-D connector.

The interconnections between the modules are set out in Table 5.3 .

Table 5.3: Table of VXI module interconnections for basic options. NC means not connected.

Connection from	To
EPC7A GPIB Interface	HP 85108A/L GPIB Interface
EPC7A GPIB Interface	AM5030 GPIB Interface
E1450A Master	E1450A Slave
E1450A ECL Trig In	50Ω termination
E1450A Ready In	NC
E1450A Marker Out	Scope Trigger (for diagnostic use)
E1450A Mini-D C0	E1340A(2) Aux In
E1450A Mini-D C1	E1340A(1) Aux In
E1450A Mini-D C2	Current E1429B(I) Ext 2
E1450A Mini-D C3	Voltage E1429B(V) Ext 2
E1450A Mini-D C4	RF Source Gate (pulse) Input
E1450A Mini-D C5	HP 85108A/L Trigger Input
E1450A Mini-D C6	NC
E1450A Mini-D C7	E1450A TTL Trig In
E1429B(I) Ch1 Lo-Z Input	AM5030 (Port 1)
E1429B(I) Ch1 Hi-Z + Input	NC
E1429B(I) Ch1 Hi-Z – Input	NC
E1429B(I) Ch2 Lo-Z Input	AM5030 (Port 2)
E1429B(I) Ch2 Hi-Z + Input	NC
E1429B(I) Ch2 Hi-Z – Input	NC
E1340A(2) Aux Out	Both E1429B Ext 1 Inputs
E1340A(2) Signal Output	E1446A(2) Input 1
E1340A(1) Aux Out	NC
E1340A(1) Signal Output	E1446A(1) Input 1
E1429B(V) Ch1 Lo-Z Input	NC
E1429B(V) Ch1 Hi-Z + Input	Port 1 voltage sense connection
E1429B(V) Ch1 Hi-Z – Input	NC
E1429B(V) Ch2 Lo-Z Input	NC
E1429B(V) Ch2 Hi-Z + Input	Port 2 voltage sense connection
E1429B(V) Ch2 Hi-Z – Input	NC
E1446A(2) Input 2	NC
E1446A(2) Diff Output	NC
E1446A(2) Main Output	Port 2 Drive
E1446A(1) Input 2	NC
E1446A(1) Diff Output	NC
E1446A(1) Main Output	Port 1 Drive

5.5 Assembly of External Hardware

5.5.1 Current Sense

Current sense can be via current probes, series resistances, or an external high-power pulser. The two measurement ports may be configured to use different types of current sense.

This subsection describes these current sensing options.

Current Probes

The dc-50MHz Tektronix model A6032 current probes are very good for this application. These are very linear and do not suffer from the common-mode injection that a series sense resistor does.

The current probes are conditioned by amplifiers that are connected to the 50Ω inputs of the HP E1429B digitizers. Consult the manuals supplied with the current probe system for assembly instructions.

Either a programmable or manual system can be assembled, depending on the choice of an AM5030 or an AM503 amplifier. The former is recommended.

AM5030 Amplifiers The AM5030 amplifiers (see Fig. 5.8) require the TM5003 power supply frame and can be controlled via a GPIB connection. Unique GPIB addresses must be assigned to each amplifier via their front panel control. One can be left at the factory default of 5, but the other must be changed, say to 6. These addresses must be entered in the Hardware Configuration File as is described in Section 5.6 .

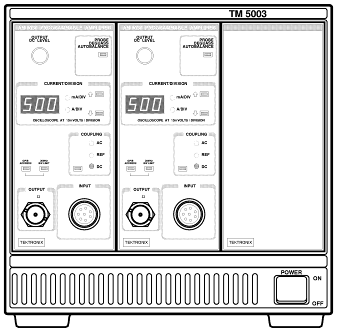

Figure 5.8: The front-panel layout of a Tektronix TM5003 power frame with two AM5030 current probe amplifiers.

AM503A Amplifiers The AM503A amplifiers cannot be remotely controlled. An appropriate entry in the Hardware Configuration File, as described in Section 5.6 , will configure to system to request manual setting of these amplifiers.

Current Sense Resistors

As an alternative to the current probes, the system will support current sense via a series resistor connected across the digitizer differential inputs. The value of the resistor used must be entered in the Hardware Configuration File (see Section 5.6). This entry will configure the system to use the digitizer differential inputs.

High-power Pulser

An external high-power pulser may provide a current sense signal that can be connected to the digitizer. The scaling factor (A/V) must be entered in the Hardware Configuration File (see Section 5.6). This entry can also be used to force the system to use the digitizer high impedance inputs.

5.5.2 RF Test Set

The APSPA system provides timing signals and GPIB support for an HP 85108A/L pulsed S-parameter test set. If used, the test set needs to be configured for external triggering and its GPIB address must be entered in the Hardware Configuration File (see Section 5.6).

The HP 85108A/L Trig input should be connected to the *C5 – RF SAMPLE* output of the APSPA timing module and the Pulse input of the gated RF source should be connected to the *C4 – RF GATE* output of the timing module.

5.5.3 External Pulser

The APSPA system provides support for an external pulser and power supply. Currently, an HP 6992A power supply with the HP 85120A K43 pulser is supported. The power supply's GPIB address needs to be entered in the Hardware Configuration File (see Section 5.6).

Channels 1 and 2 of the HP 6992A are used. Two voltage levels are provided for the bias and the pulse value. The higher potential of the two is provided by channel 2, which should be connected to the high input of the HP 85120A K43 pulser.

A TTL level control signal for the HP 85120A K43 pulser is delivered by the E1340A arbitrary function generator, and will have the correct polarity to select either the power supply's channel 1 or channel 2 voltage during the pulse period (depending on whether the pulse value is less than or greater than the bias value).

5.5.4 Computer Peripherals

The embedded controller will require a keyboard and mouse, as well as a video monitor. Consult the controller's manuals for details.

5.6 Hardware Configuration File

The instruments available to the Arbitrary Pulsed Semiconductor Parameter Analyzer, their addresses and their inter-connection details are specified in a hardware configuration file. This file has the same name as the APSPA software executable with extension '`.hcf`'. It must reside in the same directory as the APSPA executable.

A total of twelve virtual instruments make up an APSPA configuration.

- Timing/clock generator
- Port 1 (drain/collector) source/monitor unit consisting of
    - bias source,
    - pulse source,
    - current sensor,
    - voltage digitizer, and
    - amperage digitizer.
- Port 2 (gate/base) source/monitor unit consisting of
    - bias source,
    - pulse source,
    - current sensor,
    - voltage digitizer, and
    - amperage digitizer.
- RF test set.

The HCF contains the definitions of these instruments with required information about their interconnects. There is also a default set of timing calibrations for each configuration, which compensate for cable delays.

Up to ten separate hardware configurations may be defined within the file. Each corresponds to a particular "physical" configuration of the hardware. For example, two configurations are defined if both an external high-power pulser and the HP E1446A amplifier are available for driving one port of the device under test. When either is wired to the device, the appropriate configuration must be selected when running the APSPA executable.

5.6.1 Selecting a Configuration

By default, APSPA will use the first configuration that is read in. If the `-c` option is specified on the command line, another configuration other than the first one encountered may be selected.

The first configuration block is number 0, the second block is numbered 1, and so on. Specifying the `-c` option with no numeric parameter selects configuration 1 (the second configuration block). Others are specified by appending the number to the option letter, thus `-c2` specifies the third configuration encountered. (Of course, the first configuration may be specified with the option `-c0`.)

5.6.2 Format of the Hardware Configuration File

The format of the Hardware Configuration File is rigidly defined. It is not normally intended that the user alter the contents of this file, which is created when the instrument is first assembled. This information is provided to allow for system repair, or for user modification of an installed system.

Interface Names

The first line of the Hardware Configuration File must be the Interface Names Definition line with the following format:

`VXI` *vxiname* `GPIB` *gpibname*

where *vxiname* and *gpibname* are the interface names assigned respectively to the VXI bus driver and the GPIB interface driver during system software installation. Refer to system preparation in Section 2.2.1 .

This must be the first line of the file. The interface lines must be no more than seven characters and have no white spaces.

Configuration Block

Up to ten configuration blocks may be added after the Interface Names Definition line. A single configuration is defined by a block of consecutive lines within the configuration file. The first line of the block is a title enclosed in square brackets ('[' and ']'). Once a title line is started, the configuration block must be completed in a syntactically correct manner, or an error is generated. The lines in the block must be contiguous and not include blank lines.

Blank lines or text in any format may be included before, after or between configuration blocks (but not within a block). The only restriction is that the first character of any of these lines cannot be an opening square bracket, '['.

An example configuration file is shown in Figure 5.9 . The format of the lines in a configuration block is defined in Table 5.4 . Note that the parser is case sensitive.

```
VXI vxi GPIB hpib7

  This is a configuration file for APSPA version 2.6.

  The very first line of this file must have the names assigned to the VXI and GPIB
  I/O interfaces.  Use the I/0 config utility to check (or change) these names.
  The format of the first line is:
          VXI vxi_name GPIB gpib_name
  (The default names for EPconnect SICL are vxi and hpib7 respectively.)

  Free form comments, such as this one, can be added outside configuration blocks.

  The following is a basic configuration block that describes one E1446A amplifier
  per port, and a disabled RF test set: The first line of the block (text between
  brackets []) will be echoed to the user unless in quiet mode (-q option).

[One E1446A per port with AM5030 current probes... have a nice day]
E1450A 136
E1446A 88
E1340A 72
AM5030 5
E1429B 40 1
E1429B 41 1
E1446A 87
E1340A 73
AM5030 6
E1429B 40 2
E1429B 41 2
HP85108 0
TCAL -87.5e-9  -87.5e-9  -6.25e-9  0         -25e-9    0
   ---- end configuration block 0

  The following configuration block (selected by the -c of -c1 command-line option)
  describes three E1446A amplifier on Port 1, and an RF test set at address 16:

[Configured with NWA at addr 16... have a nice day]
E1450A 136
E1446A 88 3
E1340A 72
AM5030 5
E1429B 40 1
E1429B 41 1
E1446A 87
E1340A 73
AM5030 6
E1429B 40 2
E1429B 41 2
HP85108 16
TCAL -87.5e-9  -87.5e-9  -6.25e-9  0         -25e-9    0
---- end configuration block 1

  The following configuration block (selected by the -c2 command-line option)
  describes an HP 85120 K43 pulser at Port 1,

[K43 high power Port 1 drive]
E1450A 136
HP6629A 22
HP85120A 72
HP85120A 0.1
E1429B 40 1
E1429B 41 1
E1446A 87
E1340A 73
AM503A 1
E1429B 40 2
E1429B 41 2
HP85108 16
TCAL -87.5e-9  -87.5e-9  -6.25e-9  0         -25e-9    0
---- end configuration block 2
```

Figure 5.9: Example apspa.hcf hardware configuration file.

Table 5.4: Format of the lines in the hardware configuration block.

Line	Syntax	Comments
0	`[`*title text*`]`	The first character of the line must be '`[`'. The *title text* is echoed to the user prior to a measurement. Characters after '`]`' are ignored.
1	`E1450A` *ula*	Defines the timing/clock generator. Only the HP E1450A is supported. The second field, *ula*, is the instrument's address entered as a decimal integer.
2	*instr ula1 ula2*	Defines the Port 1 bias source. Legal text for field *instr* are: `E1446A` - for one or more HP E1446A amplifiers wired in parallel. The second field, *ula1*, is the primary amplifier's VXI address entered as a decimal integer. This amplifier's input impedance will be set to 50 Ω. The input impedance of the other amplifier's will be set to 1 MΩ. There are three cases for the value of the third field, *ula2*: 1. Leave the third field, *ula2*, blank to use a single amplifier. 2. Set the third field, *ula2*, to a value greater than 8 to use two HP E1446A amplifiers wired in parallel. The third field, *ula2*, is the second amplifier's VXI address entered as a decimal integer. 3. Set the third field, *ula2*, to a value less than 9 to use two or more amplifiers wired in parallel. The third field, *ula2*, is the total number of amplifier's. The other amplifiers must have consecutive VXI addresses (*ula1*+1, *ula1*+2, . . . , *ula1*+*ula2*-1). `HP6629A` - for an HP 6629A four-channel power supply, which is used in conjunction with an external high-power pulser. The second field, *ula1*, is the power supply's GPIB address entered as a decimal integer. The third field, *ula2*, is ignored and may be left blank.
3	*instr ula*	Defines the Port 1 pulse source. Legal text for field *instr* are: `E1340A` - for an HP E1340A arbitrary function generator. The second field, *ula*, is the generator's VXI address entered as a decimal integer. `HP85120A` - for an HP 85120A K43 high-power pulser, which is used in conjunction with an HP E1340A arbitrary function generator and a power supply. The second field, *ula*, is the HP E1340A arbitrary function generator's VXI address entered as a decimal integer.
4	*instr n*	Defines the Port 1 current sensor. Legal text for field *instr* are: `AM503A` - for a Tektronix AM503A manual current probe amplifier. The second field, *n*, is used to identify the amplifier when a request for manual change of range is made by the software. The value of *n* should be a decimal integer less than ten. `AM5030` - for a Tektronix AM5030 programmable current probe amplifier. The second field, *n*, is the amplifier's GPIB address entered as a decimal integer. `SERES` - for a series resistance current sensor. The second field, *n*, is the value of the sense resistor, in ohms, entered as a decimal real number. The resistance must be between 0.001 Ω and 2 MΩ. `HP85120A` - for the current sensor in the HP 85120A K43 high-power pulser. The second field, *n*, is the scale factor (Volts per Amp), which is 0.1 for the HP 85120A K43.

Table 5.4 Continued.

Line	Syntax	Comments
5	`E1429B` *ula ch*	Defines the Port 1 voltage digitizer. The only legal instrument, supported at this time, is the HP E1429B two-channel digitizer. The second field, *ula*, is the instrument's VXI address entered as a decimal integer. The third field, *ch*, is the HP E1429B channel used for this virtual instrument.
6	`E1429B` *ula ch*	Defines the Port 1 amperage digitizer. The syntax is identical to that of line 5.
7	*instr ula1 ula2*	Defines the Port 2 bias source. The syntax is identical to that of line 2.
8	*instr ula*	Defines the Port 2 pulse source. The syntax is identical to that of line 3.
9	*instr n*	Defines the Port 2 current sensor. The syntax is identical to that of line 4.
10	`E1429B` *ula ch*	Defines the Port 2 voltage digitizer. The syntax is identical to that of line 5.
11	`E1429B` *ula ch*	Defines the Port 2 amperage digitizer. The syntax is identical to that of line 6.
12	`HP85108` *gpib*	Defines the RF test set. The second field, *gpib*, is the test set's GPIB address entered as a decimal integer (enter zero if the test set will never be used).
13	`TCAL` t_1 t_2 t_3 t_4 t_5 t_6	Defines the default calibration delays for the corresponding six *set* file parameters described in Section 3.2 . The second through seventh fields are entered as decimal reals with units of seconds. t_1 delay added to Port 2 pulse source. t_2 delay added to Port 1 pulse source. t_3 delay added to amperage digitizer sample triggers. t_4 delay added to voltage digitizer sample triggers. t_5 delay added to RF test set source gate. t_6 delay added to RF test set sample trigger.

Next Step

The next step in setting up an APSPA system is installation of the software. An overview of the APSPA software and installation is given in Chapter 2 .

For more detailed information on use of the software and its operation, refer to Chapters 3 and 4 .

◇

6 User Defined Processing

6.1 Introduction

Development of a user-defined data processing code interface is planned. Contact CNERF for more details.

7 Appendix

7.1 Specifications

The APSPA system provides the necessary flexibility and capability required for both research and development as well as production line use. A 2000-point set of isothermaldynamic device characteristics can be measured in seconds.

7.1.1 Device Measurement

Quiescent Stimulus

Voltage range	±20	V
Resolution	16	bits
Current range (bias)	$> \pm 180$	mA

The bias current range can be extended to $> \pm 400$ mA by adding additional HP E1446A summing amplifiers.

Pulse amplitude specifications

Voltage range (open circuit)	±20	V
Output resistance	0.25 – 50	Ω
Pulse amplitude resolution	12	bits
Maximum peak current (<3ms)	>1	A

The peak pulse current can be extended to $> \pm 3$ A with additional HP E1446A summing amplifiers.

Pulse timing specifications

Minimum pulse width	100	ns
Maximum pulse width	25	s
Maximum pulse period	25	s
Event timing resolution	6.25	ns

Digitizer Specifications

Voltage range (1-2-5 sequence)	±0.1 – ±100	V
Current range	±0.005 – ±25	A
Resolution	12	bits
Accuracy	3	%

Measurement time (typical values)

Software and operating system overhead	∼2	s
Down load and storage time per measurement point	∼300	μs
Typical measurement time for 2000-points with 100-fold averaging and 0.1% duty cycle	3	s

Minimum theoretical measurement time is determined by pulse width, duty cycle, number of points, and averaging factor.

If a high-power pulser is used then there is an additional two second overhead each time its voltages are changed. The APSPA system minimizes the number of times this occurs.

7.1.2 Software Features

Command-line operation with event logging, scripting, and quiet mode.

Grid (start/stop/step) pulse voltage list generation.

Accepts user list of voltage points for pulses.

Maximum number of measurement points is limited by the available memory. Typically greater than 8000.

Includes timing calibration for cable length compensation.

User defined pre-measurement stabilizing period.

Pseudo-random sequencing of pulses.

Script-file command execution.

Command-line parameter override.

Software compliance checking.

Linear time-domain pulse profiling.

Logarithmic time-domain pulse profiling.

Gathers S-parameters from an HP 85108A/L system.

7.2 Pro-forma Set Files

When invoked with the `-h` or `-H` options, the APSPA software will send a pro-forma *set* file to the standard output, which can be redirected to a file. Figures 7.1 and 7.2 show the *set* files produced by software version 2.6.0.

```
#Arbitrary Pulsed Semiconductor Parameter Analyser            (AEP, JBS & JGR)
#Version 2.6.0 (May 10 1998 16:10:10)  (c) copyright by CNERF 1995,96,97,98
vdq       =      0
vdstart   =      0
vdstop    =      1
vdstep    =    0.1
vgq       =      0
vgstart   =     -1
vgstop    =      0
vgstep    =    0.1
wait      =      5
pulsetime = 1.00e-05
quiestime = 9.00e-05
tsample   = 3.50e-07
tsinset   = 1.00e-07
qpoints   =      1
average   =     16
tdinset   =      0
tqpre     = 2.00e-07
trfinset  = 1.00e-07
trftrig   =      0
nwacal    =      0
spms      =      0
idfsd     =    0.1
igfsd     =   0.01
shuffle   =      1
profile   =      0
logprof   =      0
listfile  =      0
cal       =      1
zoutg     =      0
zoutd     =      0
End
```

Figure 7.1: Template set file for APSPA generated by the `-h` option.

```
#Arbitrary Pulsed Semiconductor Parameter Analyser            (AEP, JBS & JGR)
#Version 2.6.0 (May 10 1998 16:10:10)  (c) copyright by CNERF 1995,96,97,98
vdq       =      0     # Drain source quiescent potential
vdstart   =      0     # Drain start potential
vdstop    =      1     # Drain stop potential
vdstep    =    0.1     # Drain step potential
vgq       =      0     # Gate-source quiescent potential
vgstart   =     -1     # Gate start potential
vgstop    =      0     # Gate stop potential
vgstep    =    0.1     # Gate step potential
wait      =      5     # Settling time with signal before measurement
pulsetime = 1.00e-05   # Pulse time
quiestime = 9.00e-05   # Quiescent (Q) time
tsample   = 3.50e-07   # Measurement time interval in each pulse
tsinset   = 1.00e-07   # Time from end of ’tsample’ to end of gate pulse
ppoints   =    128     # Maximum samples per measurement interval
qpoints   =      1     # Samples per quiescent time
average   =     16     # Averaging factor
tdinset   =      0     # Drain pulse inset relative to gate pulse
tqpre     = 2.00e-07   # Time of Q sample relative to start of gate pulse
trfinset  = 1.00e-07   # RF pulse inset relative to gate pulse
trftrig   =      0     # Time of NWA trigger relative to start of RF pulse
nwacal    =      0     # Pause to calibrate RF test set?
spms      =      0     # Measure S-parameters?
idfsd     =    0.1     # Drain current range FSD value
igfsd     =   0.01     # Gate current range FSD value
shuffle   =      1     # Randomise (seed=shuffle) measurement order?
profile   =      0     # Perform a time-domain pulse profile measurement
logprof   =      0     # Use log-time scale for pulse profile measurement
listfile  =      0     # Read file.LST for datapoints?
script    =      0     # Use a script file? (0=N, 1=stdin, 2=readfile/ask)
nocal     =      0     # Skip calibration of digitisers
cal       =      1     # Number of calibration iterations
zoutg     =      0     # Output impedance of GATE source
zoutd     =      0     # Output impedance of DRAIN source
timeadj   =      0     # Apply time displacements to timer outputs
tdgate    =      0     # Time displacement for gate
tddrain   =      0     # Time displacement for drain
tdsampi   =      0     # Time displacement for amperage digitiser
tdsampv   =      0     # Time displacement for voltage digitiser
tdnwa     =      0     # Time displacement for NWA RF power
tdrf      =      0     # Time displacement for RF sample trigger
End
```

Figure 7.2: Template set file for APSPA generated by the `-H` option.

7.3 Trouble Shooting

Many checks are made during each measurement to trap and report fault conditions. A log file is generated for each measurement, with progress and details of the measurement and setup. Most of the errors trapped this way are usage related. The user must address these in order to allow the APSPA system to work properly.

> Note: The E1429B digitizers will frequently display their RED error indicators during a measurement. This is normal. These are illuminated to indicate that the device has received a second ARM event from the arbitrary function generator before the requested number of samples have been taken. This occurs when the digitizers need to sample over multiple cycles of the arbitrary function generators to satisfy the requested averaging.

A common cause of incorrect data is a faulty or improper connection. For example, the digitizer may hang because the trigger cable is not connected. If the arm (Ext 1) and trigger (Ext 2) signals to the digitizer are reversed then the timing will be confused, giving nonsense data.

Another common cause of apparently incorrect data is improper setting of the timing parameters. The samples collected by the digitizers may not be properly placed within a pulse because of incorrect parameter settings or shifting of events by the timing calibration parameters. Carefully check all timing parameters and if in doubt use an oscilloscope to monitor the exact position of the pulses and trigger events.

Note that the trigger signal provided for the HP 85108A/L network analyzer must precede the desired sample time by the trigger delay set in the HP 85108A/L.

7.3.1 Error Messages

Most error messages are either self explanatory (such as "Sorry, there is not enough memory to allocate a working array. Try reducing the size of the job or closing other applications.") or should never happen, and reflect catastrophic file or system corruption or a software fault (such as "E1429B would not abort when requested at end of operation").

The following pages list all errors that the APSPA software can generate. An explanation and suggested remedy is given for each error condition. The log file generated during the measurement will contain more information that will help track down the cause of a problem.

Error codes that are not used have a blank entry.

If the problem persists, then contact CNERF.

Code	Message – possible solution – comments
0000	No Error.
0001	Sorry, there is not enough memory available. Try reducing the complexity or size of the job or closing other applications.
0002	USER REQUESTED ABORT. APSPA is attempting to terminate gracefully.
0003	Only help output was requested, no job performed. Help should be presented – not this message.
0004	Could not open a session with a VXI instrument. Check instrument address and system installation.
0005	Could not open a session with a GPIB instrument. Check instrument address and system installation. This error will occur if there is no GPIB cable or GPIB hardware.
0006	Error during closure of instrument session. This suggests a corrupted system, so try a reboot.
0007	A GPIB timeout occurred. (Consult log file.) Check GPIB cable or GPIB address of instrument. This error will occur if there is no GPIB cable or GPIB hardware.
0008	Could not map instrument session memory space. This is a low-level error. Check system installation. The instrument interface operates with system libraries that provide resources. These may be locked after a previous crash and can only be released by a bus clear or system reboot.
0009	Could not unmap instrument session memory space. This is a low-level error. Check system installation. The instrument interface operates with system libraries that provide resources. These may be locked after a previous crash and can only be released by a bus clear or system reboot.
000A	Bad instrument in the hardware configuration file. Consult the log file and repair the configuration file.
000B	A bad line in the hardware configuration file. Consult the log file and repair the configuration file.
000C	Found wrong type of instrument. Consult log file. Check ula in hardware configuration file.
000D	Hardware Configuration File could not be opened. It should be in the same directory with the same base name as this executable.
000E	Illegal hardware configuration number. Configuration number must be between 0 and 9.
000F	

Code	Message – possible solution – comments
0010	
0011	Pulse or quiescent time is negative. Check the timing parameters in the set file.
0012	Pulse event is too short or too long for the hardware. Change the timing parameters in the set file.
0013	Could not inset Port 1 pulse because an event becomes too small. Reduce the pulse inset parameter in the set file.
0014	Quiescent measurement requested too soon. Quiescent period is not long enough. Reduce the pre-pulse timing parameter in the set file.
0015	Sample period and inset exceed total pulse event time. Reduce these timing parameters in the set file.
0016	RF pulse inset relative to Port 2 pulse failed because an event becomes too small. Reduce the RF pulse inset parameter in the set file.
0017	RF trigger advance is longer than the pulse event. Shorten the RF trigger timing parameter in the set file.
0018	Pulse segments exceed pulse event period. Check the timing parameters in the set file.
0019	No pulse data requested. Check the timing parameters in the set file.
001A	
001B	
001C	
001D	
001E	
001F	

Code	Message – possible solution – comments
0020	
0021	Set file could not be opened in the current directory. Check that set file exists and is not busy.
0022	File open failed. Either file not found or is busy. Check that all files are not used by other applications.
0023	Failed to open list file, file not found or is busy. Check that list file exists.
0024	Failed to open script file, file not found or is busy. Check that script file exists.
0025	Port 1 pulse voltage is bigger than specified limit. Change the limit or the voltage request in the set file.
0026	Port 1 pulse voltage is smaller than specified limit. Change the limit or the voltage request in the set file.
0027	Port 2 pulse voltage is bigger than specified limit. Change the limit or the voltage request in the set file.
0028	Port 2 pulse voltage is smaller than specified limit. Change the limit or the voltage request in the set file.
0029	Too many pulse points requested in the set file. A numerical limit is exceeded. Reduce the size of the measurement job.
002A	No valid data lines in list (lst) file. Check for and correct the error in the list file.
002B	Second pass through list (lst) file failed. This suggests an error in the list file.
002C	
002D	
002E	
002F	

Code	Message – possible solution – comments
0030	
0031	Calibration failed, check the log file. There may be a large offset from the current probe.
0032	Error returned by user code. (For user defined processing code option only)
0033	
0034	
0035	
0036	
0037	
0038	
0039	
003A	
003B	
003C	
003D	
003E	
003F	

Code	Message – possible solution – comments
0040	
0041	E1429B SYST:ERR? returned an error. Suspect a problem with system drivers. Try rebooting. The low-level VXI bus driver may not be installed properly. There is also a hardware fault in the bus drivers of the E1429B that causes an intermittent data transfer error. A software work around for this has been used in the APSPA code. This fault is unlikely to present a problem but will produce this software error if it does.
0042	Write to E1429B failed. Suspect a problem with system drivers. Try rebooting.
0043	E1429B asked to sample too much data. This is a software fault. (`port->digitiser.maxmem`) This error may occur if user-written processing code fails to check the maximum number of samples that the digitizer can store.
0044	E1429B returned an incorrect channel tag nibble. Suspect a problem with system drivers. Try rebooting.
0045	E1429B could not be taken out of virtual instrument state. Suspect a problem with system drivers. Try rebooting. There is a hardware fault in the bus drivers of the E1429B that causes an intermittent data transfer error. A software work around for this has been used in the APSPA code. This fault is unlikely to present a problem but will produce this software error if it does.
0046	Sample count not divisible by count per pulse event. This is likely to be a software fault. This error may occur if user written processing code is faulty.
0047	E1429B data processing function initialisation failed. Check the process function or available memory. This error occurs if user written processing code has not been found.
0048	
0049	
004A	
004B	
004C	
004D	
004E	
004F	

Code	Message – possible solution – comments
0050	
0051	E1446A Amplifier output cannot be produced with current conditions. Reduce values (bias, or pulse time) in the set or list file.
0052	Amplifier input 1 is disabled Suspect a hardware fault
0053	Amplifier input 2 is disabled. Suspect a hardware fault.
0054	Peak voltage is too high for the amplifiers. Reduce the values in the set or list file. The maximum peak voltage that can be requested from an E1446A is slightly less than ±20 V. This limit may be exceeded because calibration offsets have been added to the user request. Either reduce the values of the requested voltages or turn off the calibration option.
0055	Requested output impedance is not supported. Increase `zout` in the set file, or set `zout` to zero.
0056	Too many parallel bias amplifiers requested. Check the hardware configuration file.
0057	
0058	
0059	
005A	The HP6629A was not ready within timeout period. Check the power supply operation.
005B	The HP6629A cannot deliver the requested voltage. Voltage must be GREATER than 0 and less than 50 V. The power supply cannot deliver more than fifty volts, nor can it deliver zero or negative potentials. These limits may be exceeded because calibration offsets have been added to the user request. Change the values of the requested voltages.
005C	
005D	The high-power pulser must have `zoutd` = 1 ohm. Change the setting of `zoutd` in the set file.
005E	
005F	

Code	Message – possible solution – comments
0060	
0061	AM5030 current probe amplifier is not terminated. Check connections and cycle AM5030 power (to degauss). The current probe amplifiers demand a 50Ω load, which is provided by the digitizer input. This is checked for only during a probe degaussing. The power needs to be cycled in order to instigate a degauss instruction from APSPA. Alternatively, a manual degauss can be performed.
0062	AM5030 current 'probe open' indicator is on. Close the current probe properly.
0063	AM5030 failed self test. Amplifier probably needs repair.
0064	AM5030 returned bad event code. Check connections and try cycling its power.
0065	
0066	
0067	E1340 failed to reset before timeout. Suspect a hardware fault. Try a power cycle.
0068	E1340 amplitude set too high by a software error. (Pulse amplifier gain (`pfeed`) may be too small.) This error can occur if an incompatible or faulty amplifier driver is selected in the hardware configuration file.
0069	E1340 attempted to load other than 4096 values. (This is a software fault with assembly of pulse segments.)
006A	
006B	
006C	
006D	
006E	Data arrays not allocated for loading S-parameters. This is a software error, `sarry[][]` not allocated.
006F	Cannot scan text returned by RF test set. Check the operation of the network analyser.

Code	Message – possible solution – comments
0070	
0071	Timing Module failed to reset. This suggests a hardware fault. Try a power cycle.
0072	Timing Module failed to stop on request. This suggests a hardware fault. Try a power cycle.
0073	Timing Module failed to start on request. This suggests a hardware fault. Try a power cycle.
0074	Timing Module failed to initialise before starting. This suggests a hardware fault. Try a power cycle.
0075	Stepping of Timing Module was too late. Closing other applications may fix this problem. This error occurs only when the computer's internal clock is used to pace a long pulse segment, say a bias period of several seconds.
0076	Empty timing waveform encountered by timer_load. This is a software fault. Waveform not specified.
0077	Timing waveform length exceeds total pulse period. Check times and/or calibration values in set file. The sum of the periods specified by the user for each pulse segment exceeds the overall specified pulse period.
0078	Unterminated timing sequence encountered by timer_load. This is a software fault. Waveform not specified.
0079	E1450 load gave unexpected error when tiling. Try reducing waveform complexity or number of points.
007A	Failed to fit waveform into timing module. Try reducing waveform complexity or number of points.
007B	Discontinuity in timing cycle start and end values. Try reducing waveform complexity or number of points.
007C	Timer patch cable check failure 1. Check connection or install correct cable.
007D	Timer patch cable check failure 2. Check connection or install correct cable.
007E	
007F	

Code	Message – possible solution – comments
0080	
0081	
0082	
0083	
0084	
0085	
0086	
0087	
0088	
0089	
008A	
008B	
008C	
008D	
008E	
008F	

Code	Message – possible solution – comments
0090	
0091	The total number of data points is not a multiple of pulse events or data per pulse. This indicates a software fault.(Obsoleted in 2.6)
0092	The lookup table (used for randomising pulses) is referring to more points than there are. This indicates a software fault.
0093	Target array not correctly sized. This indicates a software fault.
0094	wave_set encountered an empty user wave spec. This indicates a software fault.
0095	Empty Tim struct encountered by wave_set. This indicates a software fault.
0096	Unterminated sequence encountered by wave_set. This indicates a software fault.
0097	
0098	
0099	
009A	
009B	
009C	
009D	
009E	
009F	

◇

7.4 Revision Information

Planned Revisions

- APSPA version 3.0 to provide arbitrary sequences of pulse event segments.
- APSPA version 2.6–U to provide user-defined processing of pulse event samples.

Version 2.6.1

July 1, 1998

- Calibration no longer disables selection of 16.7Ω output impedance.
- Selecting a single point measurement with `cal`=–1 no longer crashes (divide by zero when estimating measurement total time) the program.

Version 2.6.0

May 10, 1998

- If no *set* file is found then an *scr* (script) file is opened and executed as a script.
- A new parameter *cal* has been added to set the number of calibration iterations performed.
- Improved accuracy of on-screen progress reporting.
- Cleaned up reporting in the log file.
- The user message function now looks for *ESC* or *q* keys at attempts to abort the measurement by returning a failure. Error code 0x0002 added to reflect this condition.
- An extra pre-arm segment is included to prevent excessive wait on long time settings.
- Parallel E1446A amplifiers are now mapped one at a time to conserve resources on low memory machines.
- The HP 85108A/L model C support has been improved.

Version 2.5.2

February 2, 1998

- Tab separated data fields are now generated in all output files.
- Set files can now have more than one parameter per line.
- Zero time in pulse profile is now defined and reported as being the start of the Port 2 pulse. Calibration of timing is taken into account, so that the reference point is the Port 2 connection at the DUT.

- The HP 85108A/L interface has been substantially upgraded. The network analyzer can now be set for STEP, SINGLE, or LIST sweeps and the correct frequency list is created.
- Allows exit from NWA calibration with 'Ctl-C' or 'q'.
- Checks that NWA trigger occurs during RF burst.
- Added a calibration on zero volts output to set offset, and a check at bias to set gain. (Not done if 'nocal' is not zero.)
- Script lines that have an error no longer make a measurement. Control passes to the next script line.
- Scripts accept 'exit', 'end', or 'quit', to terminate the program.
- Interactive scripting provides an opportunity to correct a typing error, rather than aborting the program.
- Setting average to zero forces only a single pulse point to be measured. Setting average to one will actually average over all samples in the 'tsample' interval.
- Can now sample after the end of the Port 2 pulse provided 'quiestime' is long enough.
- Can properly handle a request for no PIV data. (Useful for NWA measurements.)
- Doesn't report a Q point if one is not measured.
- If no pulse points are sampled, then no pulse data is output.
- The current probe is no longer reset before each measurement, so it retains its dc offset calibration.
- Fixed an error in Port 2 single point grid specification.
- Jobs with a total of between 1 and 8 samples no longer crash the digitizers.

Version 2.5.1

January 15, 1998 release.

◇