Vector Network Analysers

TRACE-HISTORY

FEBRUARY 10, 2021

ROHDE&SCHWARZ

Make ideas real

BACKGROUND

From the manual:

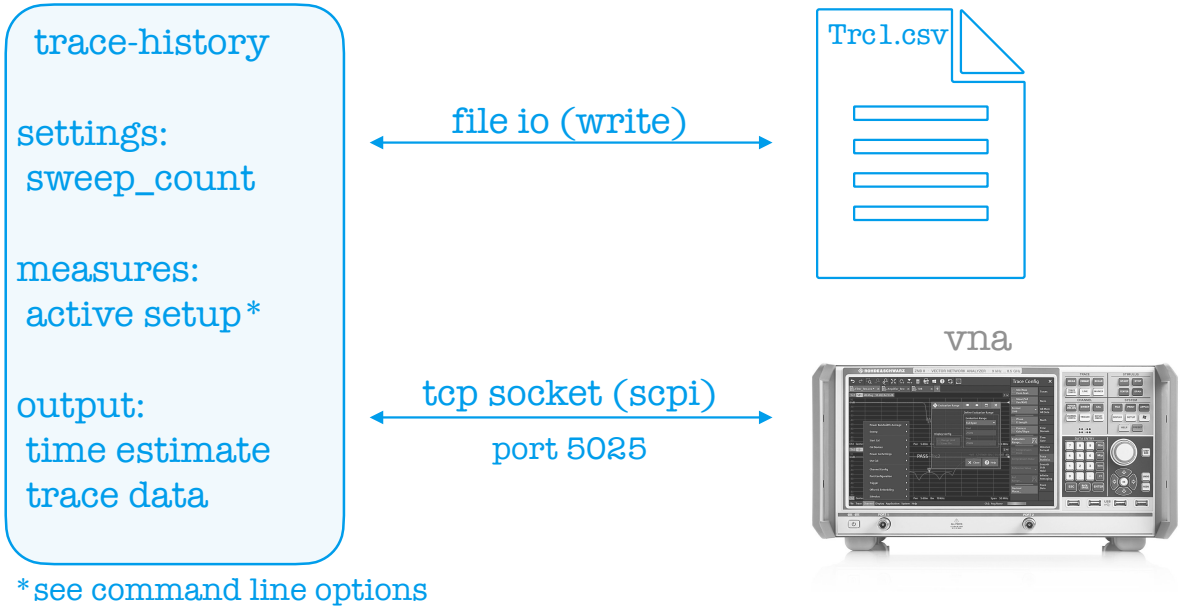
“The [trace history] commands… retrieve the results of any sweep within a previously defined single sweep group. This means that, in single sweep mode, you can first measure a specified number of sweeps and then read any of the data traces acquired.

This feature has no equivalent in manual control…” 😟

COMMAND LINE INTERFACE

```
usage: trace-history [--help]  [--quiet]  [--version]
                     [--ip-address IP_ADDRESS]  [--log-file LOG_FILE]  [--timeout-ms TIMEOUT_MS]
                     [--set-file SET_FILE]  [--data-path DATA_PATH]  sweep_count

positional arguments:
  sweep_count

optional arguments:
  --help                        show this help message and exit
  --quiet                       do not print to stdout
  --version                     show program's version number and exit
  --ip-address   IP_ADDRESS     default: localhost
  --log-file     LOG_FILE       default: vna.log
  --timeout-ms   TIMEOUT_MS     default: 120000
  --set-file     SET_FILE
  --data-path    DATA_PATH      default: current working directory
```

ARCHITECTURE



ALGORITHM: SWEEP AND TIME

- at main:line 40:

```
# sweep, measure time
vna.manual_sweep = True
vna.sweep_count  = args.sweep_count
total_time_s     = default_time_s(vna.sweep)
time_per_sweep_s = total_time_s / args.sweep_count
```

ALGORITHM: SWEEP AND TIME

- at main:line 43:

```
# sweep, measure time
vna.manual_sweep = True
vna.sweep_count  = args.sweep_count
total_time_s     = default_time_s(vna.sweep)
time_per_sweep_s = total_time_s / args.sweep_count
```

- Note: `sweep_time_s` ≅ `total_time_s` / `sweep_count` for large `sweep_count`

ALGORITHM: SWEEP AND TIME

- from default_time:line 4:

```
def default_time_s(fn):
    start_time_s = default_timer()
    fn()
    stop_time_s  = default_timer()
    return stop_time_s - start_time_s
```

ALGORITHM: SWEEP AND TIME

- from rohdeschwarz.instruments.vna.vna:line 472:
-

```
def sweep(self):
    self.manual_sweep = True
    timeout_ms = 2 * self.sweep_time_ms + 5000
    self.start_sweeps()
    self.pause(timeout_ms)
```

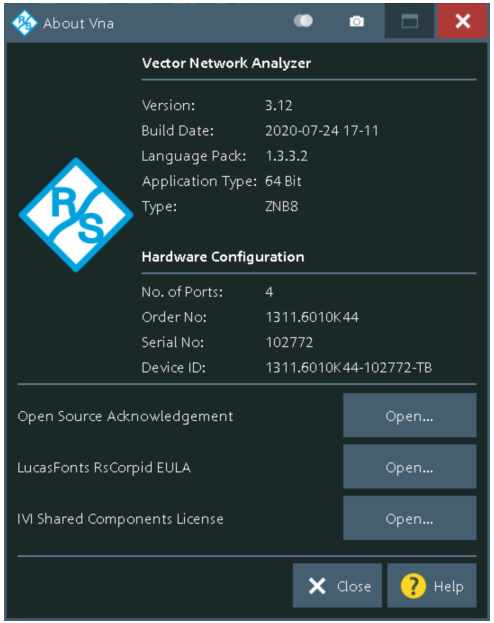
ALGORITHM: SAVE RESULTS

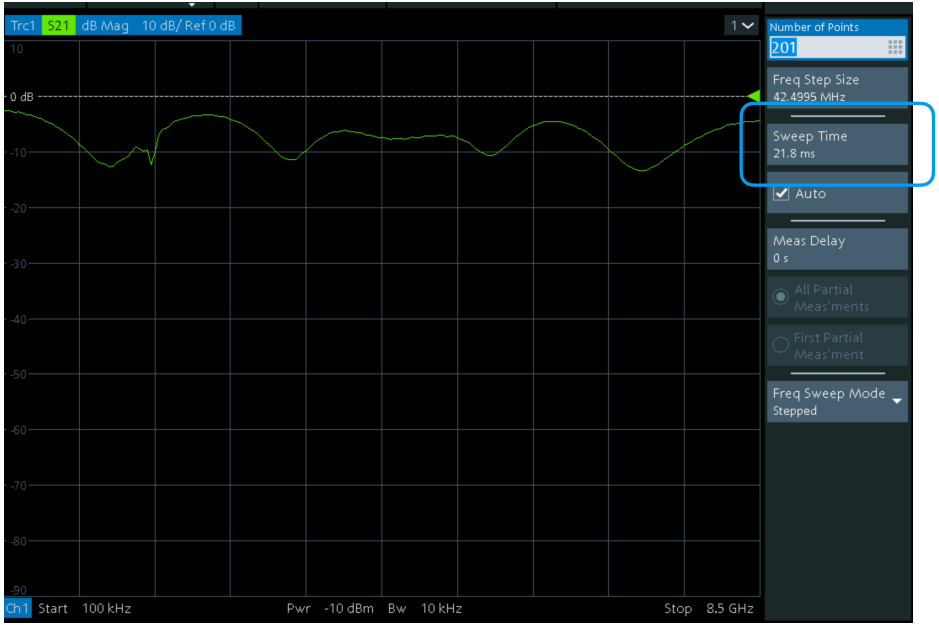
- from main: line 56:
- Note that this is occurs *after* the sweep time estimate.

```
# save trace history
for name in vna.traces:
    filename = str(data_path / f'{name}.csv')
    vna.trace(name).save_complex_history_locally(filename)
```

EXAMPLE

- Taken from the trace-history documentation
- Note: displayed sweep time (21.8 ms) is considered a *rough estimate*.





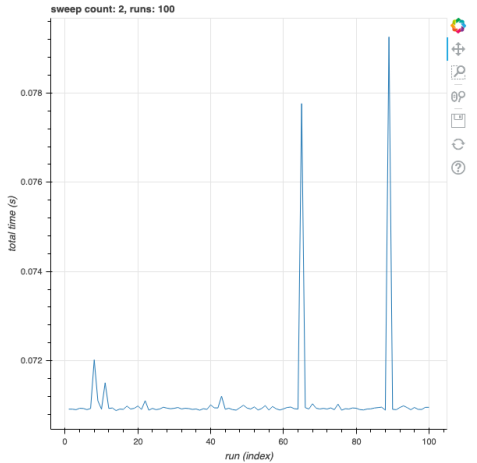
EXAMPLE

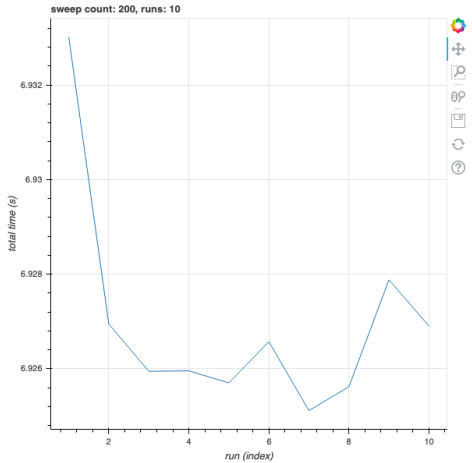
- Batch script run with SWEEP_COUNT = [2, 200, 500, 1000, 2000]
- Script runs trace-history RUN_COUNT times

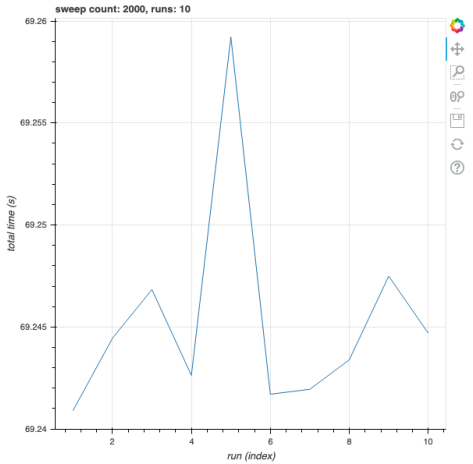
```
rem parameters
SET SWEEP_COUNT=100
SET RUN_COUNT=100

rem mkdir
SET    DATA_PATH=sweep_count_%SWEEP_COUNT%
mkdir %DATA_PATH%

rem call trace-history
for /L %%i in (1, 1, 100) do (
  trace-history\trace-history.exe --data-path %DATA_PATH% %SWEEP_COUNT%
)
```

EXAMPLE

- Uncertainty relative to mean is high at `SWEEP_COUNT=2`
- Relative uncertainty is small at `SWEEP_COUNT=200`
- Relative uncertainty is negligible at `SWEEP_COUNT=2000`







EXAMPLE

Total Time Statistics

Sweep Count	Runs	Min (ms)	Mean (ms)	Max (ms)
2	100	0.0709	0.0711	0.0793
200	10	6.9251	6.9270	6.9330
500	10	17.3107	17.3129	17.3185
1000	10	34.6193	34.6219	34.6334
2000	10	69.2592	69.2453	69.2592

Estimated Sweep Time Statistics

Sweep Count	Runs	Min (ms)	Mean (ms)	Max (ms)
2	100	35.4	35.6	39.6
200	10	34.6	34.6	34.7
500	10	34.6	34.6	34.6
1000	10	34.6	34.6	34.6
2000	10	34.6	34.6	34.6

- Note: transients are largely averaged out at `sweep_count=200`.
- Note: estimated sweep time is approximately 59% *slower* than firmware estimate.