Document Ref: EPI-163-00

Introduction

EpiSensor's ZDR supports software injection testing to simulate a frequency event so a site or asset's reaction and performance can be monitored without the need for external hardware. After initiation, the ZDR will enter injection testing mode and record high resolution data before and after a frequency event, and upload it automatically to the Gateway for visualisation and analysis.

Pre-Requisites

- Gateway Software: V04.01.00.03-RC4 or higher
- ZapHost Software: v1.04 or higher
- Gateway Hardware: NGR-30-5
- ZDR Firmware: v3.45 or higher
- ZDR Hardware: ZDR-16 with High Speed Module + GPS antenna installed w/ good satellite signal

Recording the Test Data

EpiSensor's Gateway supports exporting high resolution event data in two formats: EpiSensor JSON and Edge JSON. Before starting injection testing, configure your EpiSensor Gateway for either of these data export types, with data saved locally or exported to a remote server. Also ensure that all sensors on the ZDR have been enabled for export, which can be done by sending the following command from the Nodes List on the Gateway:
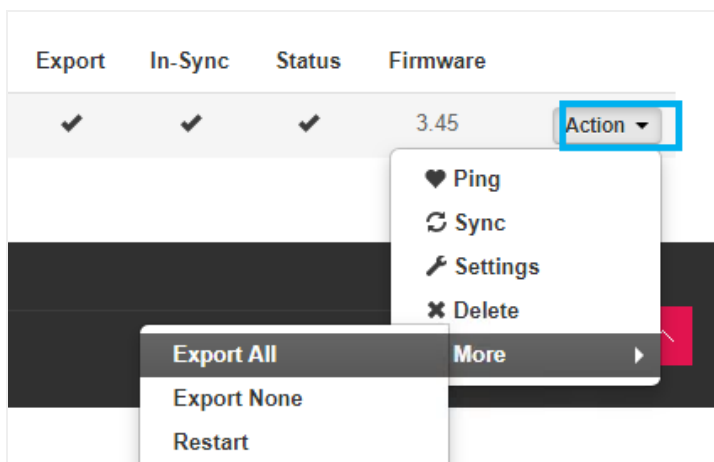
Preparing for the test

In addition to monitoring the relay state, line frequency and power, you may want to enable other sensors to report data during the test. You can do this in the norma way, by setting the reporting mode to 'snap to clock' or 'live stream' on individual sensors:

Phase A Active Power	W	61277	15:05:00 18 Jun 2019	324	✔	✔	✔	Action ▾
Phase B Active Power	W	62067	15:05:00 18 Jun 2019	325	✔	✔	✔	Action ▾
Phase C Active Power	W	57789	15:05:00 18 Jun 2019	326	✔	✔	✔	Action ▾

By default, auto exit will be set to 'false' on ZDR. If you would like the ZDR's relays to automatically return to their default state during the test, configure 'Auto Exit Event' to 'true':

Auto Exit Event	False ▾
Minimum Event Time	0	seconds
Event Reset Time	0	seconds

To enable frequency monitoring on ZDR, it is necessary to set the 'Event Sensor ID' to 'Line Frequency - 342'. This settings tells ZDR that the input it should listen to is sensor ID 342, line frequency. When this setting has been configured, two new settings will be configurable: 'Enter Event Value' and 'Exit Event Value'. These settings define the line frequency at which ZDR will actuate the relays, and the frequency threshold that must be crossed for the ZDR to exit the frequency event. Both values should be set in Hz * 1000 (i.e. a Hz value without the decimal point).

Event Sensor ID	Line Frequency - 342 ▾
Enter Event Value	49700
Exit Event Value	49800

In this example, the ZDR will enter the event and trip the relays when the frequency hits 49.7Hz, and will exit the event when the frequency returns to 49.8Hz. If the requirement for the test is different, configure these values as

appropriate. For example, if the unit should NOT enter the event at 49.7Hz, but should when it goes below this value, then set 'Enter Event Value' to 49699.

Frequency Profiles

The initial firmware release for injection testing supports one injection testing profile as follows, where the blue series represents the injected frequency in Hz, and the red series represents the relay state (low = latched).
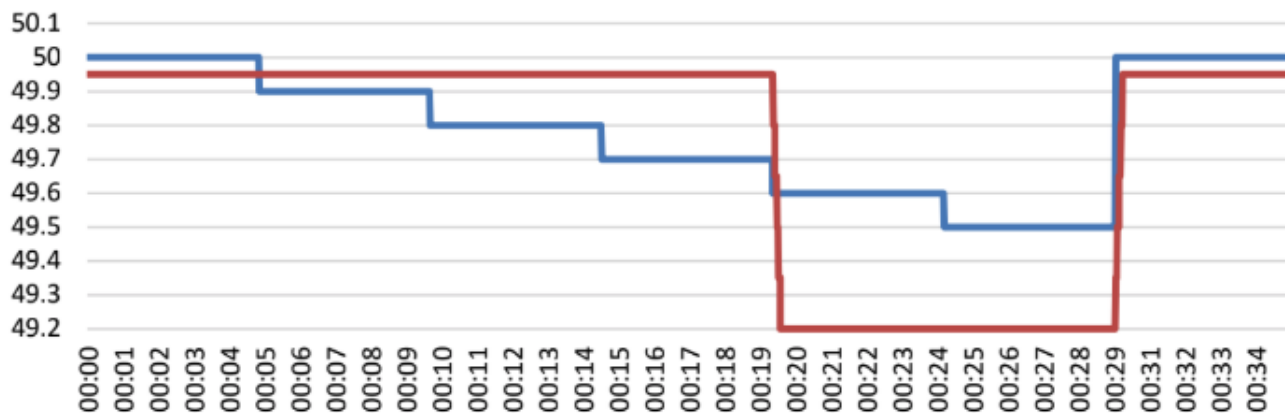

This profile will inject a line frequency as follows, and the test will last 35 minutes in total:

- 50.0Hz for 5 minutes
- 49.9Hz for 5 minutes
- 49.8Hz for 5 minutes
- 49.7Hz for 5 minutes
- 49.6Hz for 5 minutes
- 49.5Hz for 5 minutes
- 50.0Hz for 5 minutes

Initiating the test

The injection test can be initiated from the web interface of the Gateway, or with a button-press sequence on the ZDR's front panel. To start the injection test from the Gateway web interface, log in, go to the Nodes List, select the ZDR that you would like to test with, click on Action > More and 'Enable Injection Test'.
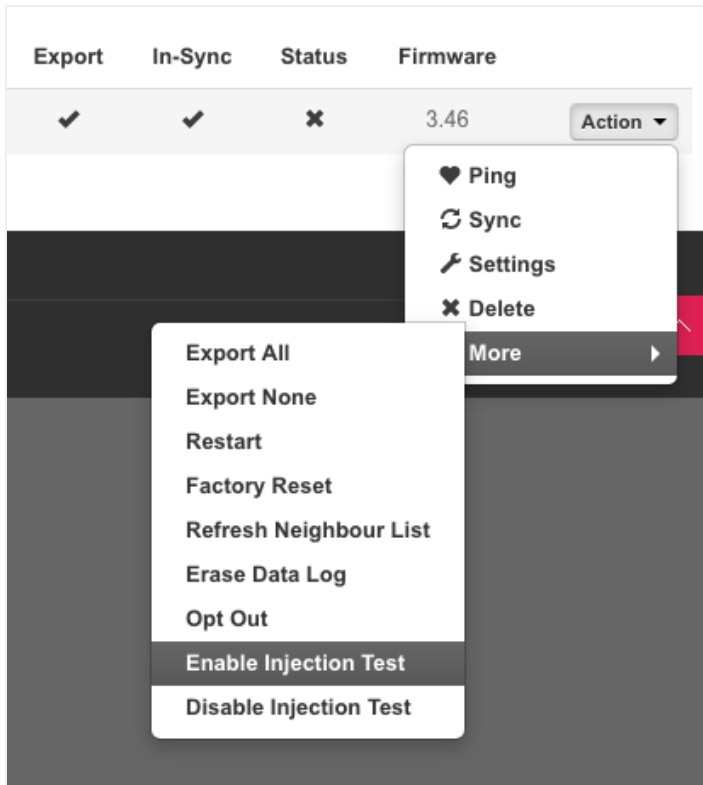
This will send a command to the ZDR, and the injection test will start on the next 15th minute.

To start the injection test from the user interface on the ZDR, press and hold the mode button for 2 seconds to activate "Command Mode". The Status LED will flicker to confirm "Command Mode" then release the button.

Next, press and release the mode button 3 more times to activate the "Testing" sub-menu. The LED will flicker in the same way as it did when entering Command Mode.

FInally, press and release the mode button 1 time to Enable Injection Test, or 2 times to Disable Injection Test. If enabled, the injection test will start on the next 15th minute.
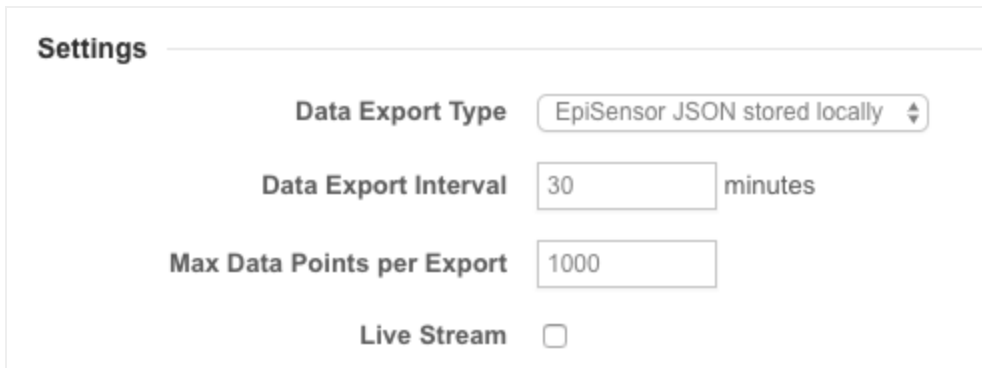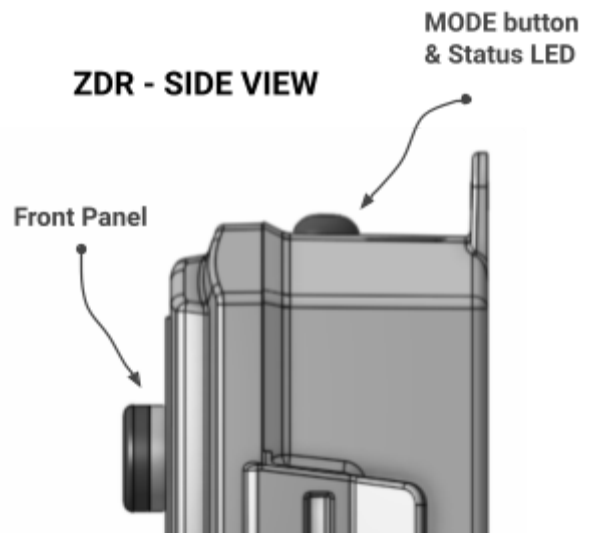
Accessing Event Data

During a frequency event, assuming the high speed data module is present, ZDR will record data at 20ms resolution for three sensors: Line Frequency (Hz), Total Active Power (kW) and Switch state (0 or 1).

In normal operation, these sensors have ID's as follows, but when recording event data, they will be exported by the Gateway with a "9" prefix before the sensor ID that's used during normal operation.

Sensor	Unit	Normal Sensor ID	Event Sensor ID
Line Frequency	Hz	342	9342
Total Active Power	kW	335	9335
Switch	0 or 1	359	9359

Note that not all data export types will export event data - only EpiSensor JSON and Edge JSON currently support this.

To store data locally on the Gateway during an event, go to the Settings > Data Export page on the Gateway and select either of the formats above 'stored locally' with 'Max Data Points per Export' set to a high value to reduce the number of data files created. Approximately 5000 data points will be produced during a typical 30 minute test.

Settings

Data Export Type	EpiSensor JSON stored locally ⬍
Data Export Interval	30 minutes
Max Data Points per Export	1000
Live Stream	☐

When the injection test has completed, data files that have been recorded will be available to download from the Data > Data File Archive page on the Gateway. Note that with the data export interval set to 30 mins, it could take up to 30 mins after the injection test has finished for all data files to be available for download.

Data File Archive

Home > Data File Archive

Data File Name	Date Last Modified	File Size	
DAT_000D6F000C5770EC_20190620142751600.json	14:29:38 20 Jun 2019 (UTC)	65 bytes	Action ▼
DAT_000D6F000C5770EC_20190620135951333.json	14:15:43 20 Jun 2019 (UTC)	280 bytes	Action ▼
DAT_000D6F000C5770EC_20190620132814399.json	13:45:43 20 Jun 2019 (UTC)	237 bytes	Action ▼

JSON data can be converted to CSV by extracting the data array from the JSON file (highlighted below) and using an online conversion tool. Note that the square brackets around the data array should be included.

```
{
  "method": "epi_edge_data",
  "ver": "1.0",
  "gateway": "000D6F000DD076E7",
  "nodes": [
    {
      "serial_number": "000D6F001023D005",
      "data": [
        {
          "id": "335",
          "export_id": "000D6F001023D005_335",
          "timestamp": "1560942480000",
          "value": "483.792"
        },
        {
          "id": "342",
          "export_id": "000D6F001023D005_342",
          "timestamp": "1560942480000",
          "value": "49.97"
        }
      ]
    }
  ]
}
```

The resulting CSV file should have four columns as follows, and can be imported to Excel for analysis.

id	export_id	timestamp	value
335	000D6F001023D005_335	1560942480000	483.792
342	000D6F001023D005_342	1560942480000	49.97