**HERO WEC LAMP Test Raw Data (TDMS Files)**

Raw data from each [LAMP](https://www.nrel.gov/water/motion-platform.html) test run of the HERO WEC can be found in the sorted [TDMS](https://www.ni.com/en/support/documentation/supplemental/06/the-ni-tdms-file-format.html) files. TDMS files are a proprietary data format by NI and are the output measurement files from the HERO WEC [MODAQ](https://nrel.github.io/MODAQ/). Each folder contains files from one test run. Within each folder is a subfolder for each DAQ measurement group containing all TDMS files recorded by that DAQ measurement group during the test run. Due to file storage concerns, each measurement group saves a TDMS file every ten minutes, limiting the maximum file size. Raw data from these files can be viewed using a TDMS viewer software such as [Scout](https://signalxtech.com/software/scout/) or Microsoft Excel [add-ins](https://www.ni.com/docs/en-US/bundle/labwindows-cvi/page/cvi/usermanual/tdmexceladdin.htm). To perform further analysis on the large volume of data collected, TDMS files are converted into [Parquet](https://parquet.apache.org/) files. The next section details the TDMS to Parquet conversion process.

***Table 1:*** *TDMS Subfolders.*

|  |  |
| --- | --- |
| **Subfolder Name** | **DAQ** |
| Anchor\_LC | Anchor load cell |
| Buoy | On-buoy MODAQ |
| CurrentAI | Onshore current transducer sensors |
| PowRaw | Generator output (electric configuration only) |
| VoltageAI | Onshore voltage transducer sensors |

**HERO WEC LAMP Test Data (Parquet Files)**

Processing used Python 3.10 and the [nptdms](https://nptdms.readthedocs.io/en/stable/) library to read all TDMS files collected during testing and prepare them to be used for data analysis in MATLAB. Using Python, TDMS files collected by each DAQ measurement group during a single test case were first concatenated in chronological order then data from all DAQs was merged into a single data frame. Since the velocity and power calculations to be performed during data analysis require all data points to use a common time stamp, Python was then used to interpolate all data onto each data group’s time array and output one data frame per data group. Each data frame contains raw data for that group’s sensors and time array along with interpolated data for all other sensors at the selected data group’s time stamps. The Pecos quality control add-in was then used to identify and replace data points outside of each sensor’s measurement range and anomalous data points. These data points were replaced with interpolated data. In cases where the encoder crossed zero during a test, raw data showed a discontinuity at each zero-crossing event and an additional Python script was used to recalculate these arrays as degrees from initial encoder position as opposed to absolute encoder position. This array was named AIN\_AIN2\_ZERO\_CROSS\_FIX\_DESPIKED. A description of each column contained in HERO WEC data Parquet files can be found in table 2 and sensor diagrams can be found in figures 1 through 4 in the MATLAB data section.

The output QC files, interpolated by group timestamp, present one complete test run in one file. In each file, relevant measurement columns are included, with the group column (indicated in the filename) representing the original timestamps. The additional columns are interpolated to provide data precisely at the group timestamp, enabling analysis of measurements across different groups. It is crucial to note that data outside of the original group is interpolated. Particularly in cases of faster sample rates, a significant volume of interpolated data from slower sample rate columns is present. For instance, consider a 5000 Hz PowRaw AIN\_AIN1 column collected at 10 Hz, where approximately 500 samples of interpolated data exist between each actual data point. While this approach simplifies analysis across measurement groups, users are advised to exercise caution and apply it selectively, especially when dealing with substantial amounts of interpolated data resulting from higher sample rates.



Figure 1 – PRESS-ON-1001 interpolated to timestamps of FLOW-ON-1001



Figure 2 – Anchor Load (LC-ST-1001) interpolated to timestamps of WEC Winch Position (AIN-2)

***Table 2:*** *Parquet File Column Descriptions.*

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Description** | **Units** |
| Timestamp\_NS | Selected DAQ timestamp | Nano-Seconds |
| Time\_Elapsed\_NS | Elapsed time | Nano-Seconds |
| IMF\_orientation\_x | IMF quaternion x orientation | Degrees |
| IMF\_orientation\_y | IMF quaternion y orientation | Degrees |
| IMF\_orientation\_z | IMF quaternion z orientation | Degrees |
| IMF\_orientation\_w | IMF quaternion w orientation | Degrees |
| IMF\_angular\_velocity\_x | IMF angular velocity about x-axis | Degrees/second |
| IMF\_angular\_velocity\_y | IMF angular velocity about y-axis | Degrees/second |
| IMF\_angular\_velocity\_z | IMF angular velocity about z-axis | Degrees/second |
| IMF\_linear\_acceleration\_x | IMF x acceleration | m/s^2 |
| IMF\_linear\_acceleration\_y | IMF y acceleration | m/s^2 |
| IMF\_linear\_acceleration\_z | IMF z acceleration | m/s^2 |
| AIN\_AIN0 | Unscaled WEC air tank pressure (PRESS\_OS\_2002) | mA |
| AIN\_AIN1 | Unscaled hydraulic pump pressure (PRESS\_OS\_1001) | mA |
| AIN\_AIN2 | Unscaled encoder position (POS\_OS\_1001) | mA |
| AIN\_AIN3 | Unscaled hydraulic pump flow rate (FLOW\_OS\_1001) | mA |
| AIN\_AIN2\_ZERO\_CROSS\_FIX\_DESPIKED | Encoder position (POS\_OS\_1001) in degrees from initial position ranging from -360 to 360. Only used when the encoder crossed zero during testing. | Degrees |
| IMU\_orientation\_x | IMU Quaternion x orientation | Degrees |
| IMU\_orientation\_y | IMU Quaternion y orientation | Degrees |
| IMU\_orientation\_z | IMU Quaternion z orientation | Degrees |
| IMU\_orientation\_w | IMU Quaternion w orientation | Degrees |
| IMU\_angular\_velocity\_x | IMU angular velocity about x-axis | Degrees/second |
| IMU\_angular\_velocity\_y | IMU angular velocity about y-axis | Degrees/second |
| IMU\_angular\_velocity\_z | IMU angular velocity about z-axis | Degrees/second |
| IMU\_linear\_acceleration\_x | IMU x acceleration  | m/s^2 |
| IMU\_linear\_acceleration\_y | IMU y acceleration | m/s^2 |
| IMU\_linear\_acceleration\_z | IMU z acceleration | m/s^2 |
| MAG\_magnetic\_field\_x | Magnetic flux in x-direction | Gauss |
| MAG\_Magnetic\_field\_y | Magnetic flux in y-direction | Gauss |
| MAG\_Magnetic\_field\_z | Magnetic flux in z-direction | Gauss |
| HDG\_Heading | WEC heading relative to true North | degrees |
| PowRaw\_PT\_ON\_1001 | Generator phase 1 voltage | Volts |
| PowRaw\_PT\_ON\_1002 | Generator phase 2 voltage | Volts |
| PowRaw\_PT\_ON\_1003 | Generator phase 3 voltage | Volts |
| PowRaw\_CT\_ON\_1001 | Generator phase 1 current | Amps |
| PowRaw\_CT\_ON\_1002 | Generator phase 2 current (not used) | Amps |
| PowRaw\_CT\_ON\_1003 | Generator phase 3 current (not used) | Amps |
| VoltageAI\_PT\_ON\_2001 | Post-rectifier voltage  | Volts |
| VoltageAI\_PT\_ON\_2002 | Charge controller input voltage | Volts |
| VoltageAI\_PT\_ON\_2003 | Charge controller output voltage | Volts |
| VoltageAI\_PT\_ON\_2004 | Battery voltage | Volts |
| VoltageAI\_PT\_ON\_2005 | Pump controller input voltage | Volts |
| VoltageAI\_PT\_ON\_2006 | Pump 1 voltage | Volts |
| VoltageAI\_PT\_ON\_2007 | Pump 2 voltage | Volts |
| VoltageAI\_CT\_ON\_2001 | Post-rectifier current | Amps |
| VoltageAI\_CT\_ON\_2002 | Charge controller input current | Amps |
| VoltageAI\_CT\_ON\_2003 | Charge controller output current | Amps |
| VoltageAI\_CT\_ON\_2004 | Battery current | Amps |
| VoltageAI\_CT\_ON\_2005 | Pump controller input current | Amps |
| VoltageAI\_CT\_ON\_2006 | Pump 1 current | Amps |
| VoltageAI\_CT\_ON\_2007 | Pump 2 current | Amps |
| VoltageAI\_PRESS\_ON\_1001 | RO inlet pressure | psi |
| VoltageAI\_PRESS\_ON\_1002 | RO post-accumulator pressure | psi |
| VoltageAI\_PRESS\_ON\_1003 | Clark pump inlet pressure | psi |
| VoltageAI\_LVL\_ON\_1001 | Permeate level (not used) | cm |
| CurrentAI\_FLOW\_ON\_1001 | RO inlet flow rate | Gpm |
| CurrentAI\_FLOW\_ON\_1002 | RO post accumulator flow rate | Gpm |
| CurrentAI\_FLOW\_ON\_1003 | Brine discharge flow rate | Gpm |
| CurrentAI\_FLOW\_ON\_1004 | Permeate flow rate | Gpm |
| CurrentAI\_CND\_ON\_1001 | Sea water conductivity | microSiemens |
| CurrentAI\_CND\_ON\_1002 | Permeate conductivity | microSiemens |
| CurrentAI\_POS\_OS\_1001 | Encoder position (when saved to onshore DAQ) | Degrees |
| CurrentAI\_FLOW\_OS\_1001 | Hydraulic pump flow rate | Gpm |
| Anchor\_LC\_LC\_ST\_1001 | Winch line tension | lbf |

**HERO WEC LAMP Test Data (MATLAB)**

Each HERO WEC data workspace is produced using the Parquet files output by the data processing Python script using the TDMS files for an individual test case. A complete list of test cases with file names, wave case information, and notes can be found in the HERO WEC LAMP Test Run Log spreadsheet. Parquet files from each test were processed using MATLAB to output a saved workspace to be used for data analysis. For each configuration of the HERO WEC, a different Parquet file was used to ensure that raw data from the most relevant sensors to each test was used for data analysis. For hydraulic and drivetrain tests, interpolation by AIN (10 Hz) data was used to produce workspaces. For electric tests, interpolation by PowRaw (5000 Hz) data was used to produce workspaces. Each data workspace includes data from each sensor used during the test case, a time array, and calculated values used to analyze component and system performance. Names of all variables which are calculated from test data begin with “Calc\_”. Due to low sample rates on the on buoy DAQ, some data needed to be smoothed before certain calculations could be performed. This was performed using MATLAB’s smoothdata function and names of all arrays of smoothed data end with “\_S”. Issues with the anchor load cell prevented winch line tension data from being saved in all cases. In these cases, a Simulink model which was calibrated from the experimental data was used to calculate a winch line tension array based on the encoder data recoded during the test, this array is named “Sim\_LC\_ST\_1001”.

***Table 3:*** *MATLAB Array Descriptions.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Array Name** | **Description** | **Subsystem** | **Units** |
| Time | DAQ time array starting at zero | MODAQ | Seconds |
| PRESS\_OS\_1001 | Hydraulic pump outlet pressure | WEC (Hydraulic) | psi |
| PRESS\_OS\_2002 | WEC air tank pressure | WEC | psi |
| POS\_OS\_1001 | Low speed spring return shaft position | WEC | Degrees |
| FLOW\_OS\_1001 | Hydraulic pump outlet flow rate | WEC (Hydraulic) | Gpm |
| PRESS\_ON\_1001 | RO system inlet pressure | RO | psi |
| PRESS\_ON\_1002 | Clark pump inlet pressure | RO | psi |
| FLOW\_ON\_1001 | RO inlet flow rate | RO | Gpm |
| FLOW\_ON\_1002 | Clark pump inlet flow rate | RO | Gpm |
| FLOW\_ON\_1003 | Brine discharge flow rate | RO | Gpm |
| FLOW\_ON\_1004 | Permeate flow rate | RO | Gpm |
| CND\_ON\_1001 | RO inlet water conductivity | RO | microSiemens |
| CND\_ON\_1002 | Permeate conductivity | RO | microSiemens |
| LC\_ST\_1001 | Anchor load cell | Anchor | Pound force |
| PT\_ON\_1001 | Generator phase 1 voltage | WEC (Electric) | Volts |
| PT\_ON\_1002 | Generator phase 2 voltage | WEC (Electric) | Volts |
| PT\_ON\_1003 | Generator phase 3 voltage | WEC (Electric) | Volts |
| CT\_ON\_1001 | Generator phase 1 current | WEC (Electric) | Amps |
| PT\_ON\_2001 | Post rectifier voltage | Power electronics | Volts |
| CT\_ON\_2001 | Post rectifier current | Power electronics | Amps |
| PT\_ON\_2002 | Charge controller input voltage | Power electronics | Volts |
| CT\_ON\_2002 | Charge controller input current | Power electronics | Amps |
| PT\_ON\_2003 | Charge controller output voltage | Power electronics | Volts |
| CT\_ON\_2003 | Charge controller output current | Power electronics | Amps |
| PT\_ON\_2004 | Battery voltage | Power electronics | Volts |
| CT\_ON\_2004 | Battery current | Power electronics | Amps |
| PT\_ON\_2005 | Pump controller input voltage | Power electronics | Volts |
| CT\_ON\_2005 | Pump controller input current | Power electronics | Amps |
| PT\_ON\_2006 | Pump 1 voltage | Power electronics | Volts |
| CT\_ON\_2006 | Pump 1 current | Power electronics | Amps |
| PT\_ON\_2007 | Pump 2 voltage | Power electronics | Volts |
| CT\_ON\_2007 | Pump 2 current | Power electronics | Amps |



***Figure 1:*** *Location of PT and CT sensors used to measure 3 phase AC electricity from generator and DC electricity after rectifier.*



***Figure 2:*** *Location of PT and CT sensors after rectifier.*



***Figure 3:*** *Location of RO system sensors.*



***Figure 4:*** *Location of on-WEC sensors.*

MATLAB was used to calculate additional data useful for analysis of the HERO WEC’s performance during testing. Encoder velocity was calculated using the forward difference method on smoothed encoder data. Winch and gearbox data was then calculated using the known gear ratios of the spring return chain drive and the gearbox. The absorbed power of the WEC was calculated using the winch velocity and winch line tension. Hydraulic power calculations were performed using the flow rate and pressure at each measured point in the system. DC electric power was calculated using the current and voltage at each measured point in the system. AC average power calculations were performed by integrating the instantaneous DC power of a single phase of the generator over the duration of the test to find total electrical work, dividing by the length of the test to find average power output of the single phase, then multiplying by 3 under the assumption that all three phases of the generator produce identical power during testing. Average power during each test was then used to calculate efficiencies and losses across various components and subsystems. Trapezoidal integration was used to calculate total volume of water moved through the system at various points over the duration of the test.

***Table 4:*** *MATLAB**Calculated and Simulated Arrays.*

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Units** |
| POS\_OS\_1001\_S | Smoothed encoder data used for winch velocity calculation | Degrees |
| Calc\_POS\_Winch | Winch position. Encoder data multiplied by first stage of spring return ratio (4.5:1) and offset by first data point in POS\_OS\_1001\_S array. | Degrees |
| Diff\_Time | Time array used to calculate velocity and power arrays (shortened by 1 data point to reflect forward difference method used to calculate velocity)Length(Diff\_Time) = Length(Time) - 1 | Seconds |
| Calc\_V\_Encoder | Encoder velocity calculated using forward difference method (see Diff\_Time) | rpm |
| Calc\_V\_Winch | Winch velocity. Calc\_V\_Encoder multiplied by 4.5 (first stage spring return ratio) | rpm |
| Calc\_V\_Gearbox | Gearbox output shaft velocity. Calc\_V\_Winch multiplied by 11.28 (gearbox ratio) | rpm |
| Calc\_P\_abs | Instantaneous absorbed power (Calc\_V\_Winch \* LC\_ST\_1001) | kW |
| Calc\_P\_WEC\_Electric\_Out | Instantaneous power output from generator | kW |
| Calc\_Eta\_WEC\_Electric | Drivetrain efficiency during test (winch to generator) | % |
| Calc\_P\_Electric\_1 | Instantaneous post-rectifier electric power | kW |
| Calc\_Eta\_Rectifier\_Instantaneous | Instantaneous rectifier efficiency | % |
| Calc\_P\_Electric\_2 | Instantaneous charge controller input power | kW |
| Calc\_P\_Electric\_3 | Instantaneous charge controller output power | kW |
| Calc\_Eta\_CC\_Instantaneous | Instantaneous charge controller efficiency | % |
| Calc\_P\_Electric\_4 | Instantaneous power into batteries | kW |
| Calc\_P\_Electric\_5 | Instantaneous power supplied to pump controllers | kW |
| Calc\_P\_Electric\_P1 | Instantaneous power supplied to pump 1 | kW |
| Calc\_P\_Electric\_P2 | Instantaneous power supplied to pump 2 | kW |
| Calc\_Filter\_DP | Pressure differential across RO pre-filter | psi |
| Calc\_ACC\_Flow | Flow rate into onshore accumulators | Gpm |
| Calc\_P\_Hydraulic\_1 | Hydraulic power at pump outlet | kW |
| Calc\_P\_Hydraulic\_2 | Hydraulic power at RO system inlet | kW |
| Calc\_P\_Hydraulic\_3 | Hydraulic power at Clark pump inlet | kW |
| Calc\_Recovery\_Ratio | RO recovery ratio during test |  - |
| Sim\_LC\_ST\_1001 | Instantaneous winch line tension calculated using Simulink. Only used in test cases where anchor load cell data was not recorded. | Pound force |

***Table 5:*** *MATLAB Calculated and Simulated Values*

|  |  |  |
| --- | --- | --- |
| Calc\_Tension\_AVG | Average load cell output (i.e. tension) during test | Pound force |
| Calc\_Tension\_Max | Maximum load cell output during test | Pound force |
| Start.Index | Index of first data point in LAMP test profile where motion initiates |  - |
| Start.Time | Time value of Start.Index | Seconds |
| End.Index | Index of last data point in LAMP test profile (end of hold period) |  - |
| End.Time | Time value of End.Index | Seconds |
| Calc\_P\_abs\_AVG | Average absorbed power during test | kW |
| Calc\_Q\_Pump\_AVG | Average flow rate supplied by hydraulic pump during test | Gpm |
| Calc\_P\_WEC\_Electric\_Out\_AVG | Average power output of generator during test | kW |
| Calc\_P\_Electric\_1\_AVG | Average post-rectifier electric power during test | kW |
| Calc\_Eta\_Rectifier | Rectifier efficiency during test | % |
| Calc\_P\_Electric\_2\_AVG | Average charge controller input power during test | kW |
| Calc\_P\_Electric\_3\_AVG | Average charge controller output power during test | kW |
| Calc\_Eta\_CC | Charge controller efficiency during test | % |
| Calc\_Eta\_Electric\_Overall | Overall electrical system efficiency during test (winch to charge controller output) | % |
| Calc\_P\_Electric\_4\_AVG | Average power into batteries during test | kW |
| Calc\_Battery\_Charge | Calculated energy input to battery over duration of test (positive value indicates battery charge, negative value indicates battery discharge) | kWh |
| Calc\_P\_Electric\_5\_AVG | Average power supplied to pump controllers during test | kW |
| Calc\_P\_Electric\_P1\_AVG | Average power supplied to pump 1 during test | kW |
| Calc\_P\_Electric\_P2\_AVG | Average power supplied to pump 2 during test | kW |
| Calc\_Loss\_Conversion | Conversion loss at electric pumps |  - |
| Calc\_Q\_Feed\_AVG | Average flow rate into RO system during test | Gpm |
| Calc\_Volume\_Pump | Total volume of water moved by hydraulic pump during test (in the electric configuration with RO on, this is equivalent to Calc\_Volume\_RO) | Gallons |
| Calc\_Volume\_RO | Total volume of water received at RO | Gallons |
| Calc\_Volume\_PRV | Total volume of water discharged by pressure relief valves during test. Calc\_Volume\_Pump – Calc\_Volume\_RO | Gallons |
| Calc\_Volume\_Permeate | Total volume of fresh water produced during test | Gallons |
| Calc\_PRV\_Percent | Percent of water pumped that is discharged by pressure relief valves during test | % |
| Calc\_P\_Hydraulic\_1\_AVG | Average hydraulic power at pump outlet during test | kW |
| Calc\_P\_Hydraulic\_2\_AVG | Average hydraulic power at RO system inlet during test | kW |
| Calc\_P\_Hydraulic\_3\_AVG | Average hydraulic power at Clark pump inlet during test | kW |
| Calc\_Loss\_Filter | Percent of energy loss across RO pre-filter | % |
| Calc\_Eta\_Overall | Overall system efficiency during test (winch to Clark pump) | % |
| Calc\_Q\_Permeate\_AVG | Average permeate flow rate during test | Gpm |
| Calc\_Loss\_Transfer | Percent of energy loss across hydraulic transfer hose | % |
| Sim\_Tension\_AVG | Average winch line tension calculated using Simulink | Pound force |
| Sim\_Tension\_Max | Maximum winch line tension calculated using Simulink | Pound force |
| Run.Time | End.Time – Start.Time  | Seconds |
| Calc\_Eta\_WEC\_Hydraulic | Drivetrain efficiency in hydraulic configuration during test (winch to pump or output) | % |
| Calc\_Eta\_WEC\_Electric | Drivetrain efficiency in electric configuration during test (winch to generator output) | % |