LoggedBehavior

Factory

ElectricalInterface

_driving_terminal_amps
_driving_terminal_volts
_control

drive_current(current, control)
drive_voltage(voltage, control)
sample_current()
sample_voltage()

respond_to_control_changes

For Testing:
Sample simulator here

respond_to_control_changes

entry /
  self.publish(
    signal=signals.SET_CURRENT_SAMPLER
    payload=Sampler(fn=partial(self.sample_current)))

  self.publish(
    signal=signals.SET_VOLTAGE_SAMPLER
    payload=Sampler(fn=partial(self.sample_voltage)))

Sampler = namedtuple(
  'Sampler', ['fn'])

self.subscribe(Event(signal=signals.REQUEST_FOR_SAMPLERS))
self.subscribe(Event(signal=signals.DRIVE_CURRENT))
self.subscribe(Event(signal=signals.DRIVE_VOLTAGE))

DRIVE_CURRENT as e /
  self._driving_terminal_volts = None
  self._driving_terminal_amps = e.payload.amps

DriveCurrent = namedtuple(
  'DriveCurrent', ['amps', 'control', 'sec'])

drive_current_state

entry /
  self.drive_current(
    self._driving_terminal_amps
    self._control
)

DRIVE_VOLTAGE as e /
  self._driving_terminal_amps = None
  self._driving_terminal_volts = e.payload.volts

DriveVoltage = namedtuple(
  'DriveVoltage', ['volts', 'control', 'sec'])

drive_voltage_state

entry /
  self.drive_voltage(
    self._driving_terminal_volts,
    self._control
)

For Testing:
Run simulator here

REQUEST_FOR_SAMPLERS