

entry / self.control.reference = \ self.battery_spec.abs_ref_volts self.start_sec = self.sec self.post_fifo(Event(signal=signals.electrical_change)) self.cause = self.state_name Tick as e / if e.payload.sec - self.start_sec > \ self.battery_spec.abs_timeout_sec or self.amps > self.battery_spec.abs_exit_amps: self.post_fifo(Event(signal=signals.To_Float))

entry / self.control.reference = \ self.battery_spec.float_ref_volts self.post_fifo(Event(signal=signals.electrical_change)) self.cause = self.state_name

```
entry /
self.control.reference = \
self.battery_spec.equ_ref_volts
self.start_sec = self.sec
self.post_fifo(
Event(signal=signals.electrical_change))
self.cause = self.state_name

Tick as e /
if e.payload.sec - self.start_sec > \
self.battery_spec.equ_timeout_sec:
self.post_fifo(Event(signal=signals.To_Float))
```

