Battery Cycling Test and Automation Considerations

APPLICATION NOTE

KEITHLEY
A Tektronix Company

Tektronix®

Introduction

The worldwide secondary (rechargeable) battery market is poised for continued and tremendous growth, due in large part to the ever-increasing demands on capacity, life, efficiency, and power delivery capability. The list of battery applications includes smart phones, tablets, laptops, portable speakers, wireless headphones, smart watches, and electric vehicles. Batteries are used in a variety of consumer and industrial applications, so their testing requirements typically depend on their chemistry, size, and specific uses. While smaller consumer electronics tend to rely on single battery cells, larger devices require a battery module (a collection of cells configured in series and parallel combination) or battery packs (a series of modules inclusive of a battery management system).

Rechargeable batteries are routinely tested through battery cycling, a combination of charge and discharge routines that are repeated as needed for the test. The initial cycle is a critical step in the battery manufacturing process, where the solid electrolyte interphase layer is formed. Subsequent cycles age the battery, before it is judged for quality, efficiency and capacity. Continued cycling also helps quantify the expected number of cycles the battery can undergo before capacity degradation sets in.

A typical battery cycling test set-up may include programmable power supplies, electronic loads, voltmeters, and ammeters or an instrument that provides a mix of features from all four instruments, such as a source measure unit.

This application note describes charge and discharge methods as well as applied pulsed current to better represent real world load scenarios. Example programs for automating battery cycling phases are also provided.

Test Description

Secondary batteries may be charged/discharged using a variety of methods, depending on the application or the demands of certain standards. For battery cycling of rechargeables, the first phase of the test involves charging the battery where energy is forced into the device until it reaches its energy storage potential. During this process, the battery voltage and energy delivered is monitored to determine the device state of charge (SoC). Following this is the discharge phase where energy is removed from the device to the point where its voltage is at or below a usable level. Again, the battery voltage and energy dissipated is monitored, this time to determine the device depth of discharge (DoD). Depending on the battery application and the standards that may govern it, a wait or hold time may be enforced between the charge and discharge phases.

The rate at which energy is added or removed from the battery depends on the cycling method used. Batteries can be charged using either a constant current (CC) or a constant voltage (CV) method. When the CC method is used, a programmable current source supplies a constant current. The current source will continue discharging or charging the battery until it reaches a preprogrammed voltage level.

The current levels used for the CC method are specified in terms of the battery's capacity (C). The capacity of the battery is defined as the time integral of the current flow out of the battery from the beginning of the current flow (t=0) to a time when it reaches a specified cut-off voltage, and can be expressed as:

$$\text{Capacity} = \int \Delta t i dt$$

The capacity is specified in terms of ampere-hours and should be expressed in terms of a load current. The charge rates used vary depending on the application, but many standards and datasheets will define them in terms of a C rate, where 1C is the current required to discharge the entire battery in 1 hour. This allows rates to be normalized regardless of a particular battery's capacity or current ratings. For example, a 500 milliampere-hour cell, discharged at 50 mA, is discharged at the 0.1C or C/10 rate. This battery, with a capacity of 500 mAh @ 50 mA, could supply a 10 mA load for 50 hours. Factors affecting capacity include battery size, chemistry, temperature, discharge rate, open circuit time, etc.

In the CV method, a programmable voltage source supplies a fixed voltage that is equivalent to the voltage rating of the battery. The current of the voltage source is limited to a safe charging rate. As the battery becomes fully charged, the current will decrease until it reaches zero or near zero.

In some cases, it is preferred to combine the CC and CV methods to ensure ideal state conditions for the battery device under test.

To prevent safety hazards or damage to the battery, care must be taken not to overcharge the battery. Additionally, discharging a battery to levels below its specified cutoff voltage rating can result in longer charge time or may damage the battery.

Test Procedures

Charging/Discharging Using the Constant Current Method

The preferred and dominant method for charging and discharging batteries is to do so using a constant current. The configuration for this scenario is illustrated below.

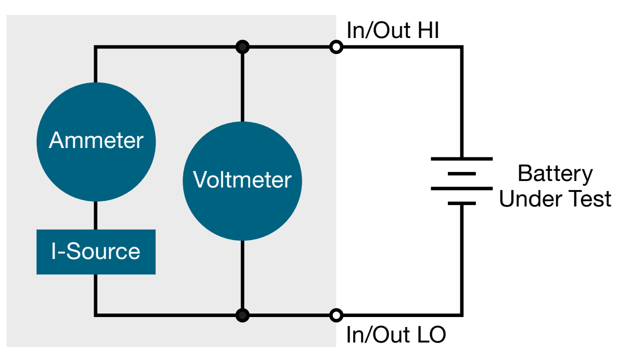

Figure 1: Instrument arrangement for conducting constant current battery cycling.

As shown in **Figure 1**, a current source is either delivering or removing charge to the battery while an ammeter and voltmeter are used to measure current or voltage, respectively. The source is set up by first selecting the proper current output value. When charging, a positive test current is used; when discharging, a negative test current is used. The setting of the voltage limit (or compliance) will depend on whether the battery is being charged or discharged.

For charging only, the limit voltage should be set to the desired battery voltage level. Depending on the charge needs or specifications, this could be the nominal voltage (the expected full battery potential under load) or the open circuit voltage, or a percentage of either of these target levels. Once the source output is turned on, the instrument will output a constant current until it reaches the target voltage limit level.

For discharging, the limit voltage should always be set higher than the nominal voltage of the battery. It should never be set lower than the battery voltage because the instrument will go into a limit condition (sometimes referred to as compliance), causing the output current to be much higher than desired. The battery voltage is measured by the voltmeter instrument, then compared to the desired threshold level. Once the threshold level is achieved, the current source is turned off.

Due to its versatility, a source measure unit (SMU) can be used to accomplish the functionality of current source, ammeter, and voltmeter shown in **Figure 1**. The SMU also tends to have better accuracy specifications and more measure ranges to work with in comparison to more basic power supply. Further, in addition to the source measure unit acting as a source, it can also sink current and act as an electronic load. Take caution when using an SMU for sourcing current to charge and/or discharge batteries, making sure the battery voltage never exceeds the voltage compliance setting because this will cause excessive current to be drawn from the battery under test. Also, if the feature is available, be certain the output-off state of the current source is set for high-impedance (or HI-Z). This setting opens the output relay when the output is turned off. Under normal output-off state conditions (typically the default condition), turning the output off may set the voltage compliance to zero. This zero-volt compliance condition will also cause excessive current to be drawn from the external battery.

Alternatively, a power supply may be used for the current source during the charge phase and an electronic load may be used for the discharge phase. However, power supplies generally operate in constant voltage mode and must be driven to a constant current condition, a setup which might prove less than ideal for the test. The current and voltage readback features of either the power supply or the electronic load may be used to act as the ammeter and voltmeter, though may yield less accurate measurements than desired. If this is the case, the operator can always supplement either or both the ammeter and voltmeter with a precision digital multimeter. For situations where the power supply or electronic load has no HI-Z feature, yet it is imperative to isolate their terminals from impacting the battery when the outputs are off, external relays can be inserted between the outputs and the battery.

Charging/Discharging Using the Constant Voltage Method

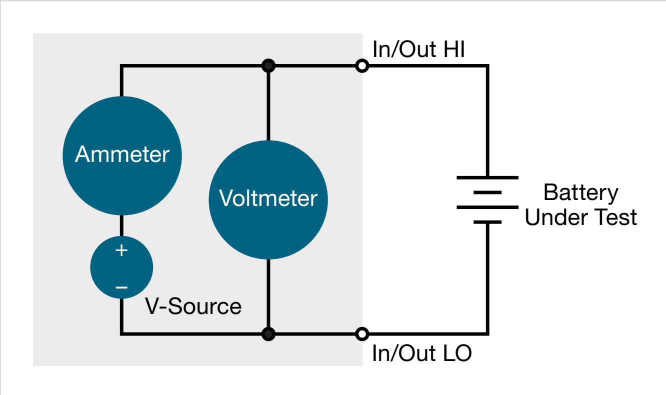

Figure 2: Instrument arrangement for conducting constant voltage battery cycling.

Figure 2 illustrates a battery connected to instrumentation intended to apply the constant voltage method for charging/ discharging. The voltage source is either delivering or removing charge to the battery while an ammeter and voltmeter are used to measure current or voltage, respectively. The voltage source value is set to the desired level to which the battery will ultimately be charged or discharged. The limit (or compliance) current is set at an appropriate level at which to charge and/or discharge for that battery type. As the instrument is either charging or discharging the battery to the desired voltage level, the instrument will be in current limit (compliance) until it reaches the programmed voltage level. While the instrument is in current limit, it is actually acting as a constant current source.

Charging Using a Combined Constant Current, Constant Voltage Method

Batteries will often be found to specify a hybrid method for at least charging where both CC and CV source priorities are involved. The battery will be charged under constant current conditions up to the point where the specified voltage limit is reached. At this point, the mode is changed to constant voltage and the current is monitored. Once the current reaches a certain level, the battery is considered charged. For example, one lithium-ion battery datasheet may indicate to charge to 4.2 V at a constant current rate of 0.5C. Upon reaching the limit, the method is changed to CV where 4.2 V is the applied source, and the current is then measured until it achieves a level of approximately 0.01C of the battery capacity.

The SMU is the perfect choice for applying the hybrid charging method since it operates as either voltage supply or a true current source. Further, the programming capability allows the operator to transition between the modes almost seamlessly. The example shared in Appendix A shows the script setup and operation for the hybrid charge method as it would be applied using a 2400 Graphical Series SMU.

Automating the Charge and Discharge Cycles

Charge and discharge cycles often take several hours to complete. Further, standards and specifications may indicate various charge/discharge rates (0.1C, 0.2, 0.5C, 1C, 2C, and 5C to name just a few), possibly interleaving these as well as applying different delay times between cycles. A rated charged test may only call for five cycles to be performed whereas a cycle life test may call for greater than three hundred cycles to be performed. Therefore, automating these tests is desirable. Because most modern instrumentation includes any of a number of interface options (LAN, USB, GPIB, or RS-232), the charge/discharge cycling of the batteries can be automated by connecting the instrument to a personal computer and issuing commands from the instrument's command set.

To share examples of what some automation options might be available, the appendix of this application note provides TSP script code which can be executed directly on any one of Keithley 2400 Graphical Series SMUs without the need of a connected PC. The first script (Appendix A) offers a look at how to apply the hybrid CC/CV charging method to an 18650 battery, first applying a 0.2C current source until the 4.2 V threshold is reached. The source then changes to CV and current is monitors until it drops below a target level.

The second script (Appendix B) guides the operator through the setup of the instrument to discharge an 18650 lithium ion battery using the CV method, typically down to 2.5 V using a 0.2C load or rate. One voltage reading is taken every two seconds, which is then stored in a data file, along with a corresponding timestamp reading. This stored data makes it possible to plot voltage vs. time for the discharge cycle. As the voltage readings are being taken, they are compared with the 2.5 V threshold level. When this level is reached, the output is turned off.

Figure 3 shows an example plot of data taken while the battery was being discharged using a CV method, the script for which can be found in Appendix C. Both the voltage and time readings were generated using the source measure unit. The data was plotted directly on the instrument and shown via the Graph view.

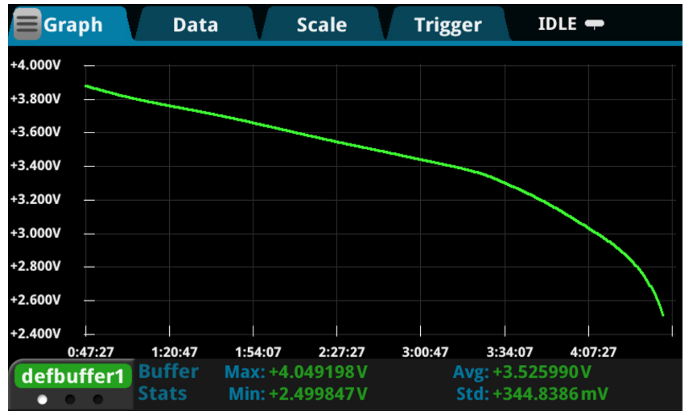

Figure 3: Data plotted while applying a constant voltage discharge to a lithium ion battery.

Discharge Considerations Using a Pulsed Current

The primary focus of this application note to this point has been with respect to cycling using a constant source, either current or voltage. However, real world device usage regularly – if not predominantly – deviates from any truly predictable power draw. We see this in numerous connected or wireless devices that include some form of RF transmission, often transitioning between states of somewhat consistent power consumption in general use to intermittent high-power use as data is transmitted to and from the device. Examples of this might be something as simple as a small Bluetooth device operating from a coin cell battery to a smart phone performing any number of a multitude of tasks using a lithium-ion battery. The same can be said of electric vehicles where the battery packs are stressed with different load currents depending on the acceleration rate the driver imposes on the EV during a drive to work.

To account for this type of irregular usage, battery manufacturers often include pulsed discharge specifications in addition to their constant discharge specifications, calling out certain durations and intervals over which the battery load should fluctuate to offer better insight into real-world usage expectations. Take for instance a common CR2032 coin cell whose pulse discharge characteristics indicate that a 2 s pulse is to be applied twelve times per day for however many days it takes to reach the cutoff voltage. Likewise, battery modules and packs used in hybrid and electric vehicles may apply pulse performance characteristic testing involving a series of discharge rates for any number of seconds, regenerative (charge) rates for some number of seconds, and rest periods in minutes) in between. While the pulse durations indicated above do not seem all that aggressive, other devices may specify pulse load intervals in milli- or microseconds.

Pulsed waveforms can be created by the Series 2400 SourceMeter by either changing the level of the fixed source output or by using either configuration lists or source list commands, depending on the model selected. These list setup features are used if the desired pulsed output is less than about one second. If the desired pulsed output is greater than one second in length, it is better simply to change the output using the standard fixed source output commands.

Summary

As the market grows and new types and chemistries emerge, understanding and studying capacity and life of batteries will continue to be a necessity. Therefore, education on the different nuances of the charge and discharge phases of a battery cycling test will be beneficial to engineers and technicians alike, preparing them for roles in either research and development or production testing. An example of a full cycling TSP script is included in Appendix C and can be executed directly on the instrument without the need for a PC.

Whether conducting the charge or the discharge phase, it is important to be sensitive to the applied method to best represent how the battery will be operated. In most cases, the constant current method is preferred for both, while in others the constant voltage method is sufficient. For charging, it is not uncommon for find that a hybrid approach will help yield the best overall performance.

While constant conditions have historically been the norm, battery manufacturers are now more sensitive to the impact of real-world applications where load conditions fluctuate while the battery is in use. Test standards and battery datasheets may now include details on methods for pulse testing, where the device is subjected to periods of higher current draw.

Sections of this application note directly point to the use of source measure units to conduct the cycling tests, but these tests can be accomplished using the pairing of power supplies and electronic loads. In fact, with power supplies and electronic loads offering a greater power envelope, they may be better positioned to serve the application needs when testing battery module and pack configurations. If measurement accuracy and resolution are a concern, digital multimeters can supplement the test configuration.

Appendix A: TSP Code Used to Apply a Hybrid Charge Cycle to an 18650 Lithium-Ion Battery

```
-- This code charges a 3400 mAh 18650 lithium-ion battery to its full
-- potential using a hybrid constant current, constant voltage method.
-- The initial charge rate is 0.2C (for 680 mA) which is applied until
-- the battery reaches the threshold voltage. Upon reaching this target,
-- the source mode is changed, and 4.2 V is applied with a 680 mA limit.
-- The source output remains on until the measured current drops below
-- 0.05C (170 mA).
reset()
local capacity = 3.400  -- in Ah
local crate = 0.2
local currlevel = capacity * crate
local thresholdv = 4.2
-- Establish settings and configuration lists applicable to
-- CC and CV modes of operation.
-- Start with the CC source configuration
smu.source.func = smu.FUNC_DC_CURRENT
smu.source.range = currlevel     -- will apply appropriate source range
smu.source.vlimit.level = thresholdv
smu.source.offmode = smu.OFFMODE_HIGHZ
smu.source.level = currlevel
smu.source.configlist.create(“sourcelist1”)
smu.source.configlist.store(“sourcelist1”, 1)
-- Next add the measure configuration
smu.measure.func = smu.FUNC_DC_VOLTAGE
smu.measure.sense = smu.SENSE_4WIRE
smu.measure.range = thresholdv      -- will apply appropriate measure range
smu.measure.configlist.create(“measurelist1”)
smu.measure.configlist.store(“measurelist1”, 1)
-- Follow up with the CV source configuration
smu.source.func = smu.FUNC_DC_VOLTAGE
smu.source.offmode = smu.OFFMODE_HIGHZ
smu.source.range = thresholdv
smu.source.ilimit.level = currlevel
smu.source.level = thresholdv
smu.source.configlist.store(“sourcelist1”, 2)
-- Then add the complementary measure configuration
smu.measure.func = smu.FUNC_DC_CURRENT
smu.measure.sense = smu.SENSE_4WIRE
smu.measure.range = currlevel
smu.measure.configlist.store(“measurelist1”, 2)
-- Recall the CC source/measure configuration
smu.source.configlist.recall(“sourcelist1”, 1)
smu.measure.configlist.recall(“measurelist1”, 1)
-- Turn the output of the SMU on
smu.source.output = smu.ON
-- Loop voltage measurements at 2 s intervals until the
-- threshold voltage is reached (+/-0.5%)
local samplev = smu.measure.read()
local stoppoint = thresholdv - (thresholdv * 0.005)
while samplev < stoppoint do
   delay(2.0)  -- measure every 2 s
   samplev = smu.measure.read()
   print(samplev)
end
-- Once the threshold is achieved, change to CV sourcing
smu.source.configlist.recall(“sourcelist1”, 2)
smu.measure.configlist.recall(“measurelist1”, 2)
smu.source.output = smu.ON -- is this even necessary???
-- Now loop until the current is <= 0.05C
local cutoffcurrrate = 0.05
local cvcutoffcurr = capacity * cutoffcurrrate
local samplei = smu.measure.read()
while samplei > cvcutoffcurr do
   delay(2.0)  -- measure every 2 s
   samplei = smu.measure.read()
   print(samplei)
end
-- Turn the output of the SMU off
smu.source.output = smu.OFF
```

Appendix B: TSP Code Used to Apply a CV Discharge Cycle to an 18650 Lithium-Ion Battery

```
--This code discharges a 3400 mAH 18650 battery to 2.5 V with a
--discharge current of 0.68 A (0.2C).
--Prior to executing code, reset instrument from front panel and
--set the Output Off state of the SMU to the High Z Mode
--Insert USB drive to save readings
--clear the buffer
defbuffer1.clear()
local currlimit = 0.68
local batterycutoffv = 2.5
--Measurement Settings
smu.measure.func = smu.FUNC_DC_CURRENT
smu.measure.range = currlimit
smu.measure.sense = smu.SENSE_4WIRE
--Source Settings
smu.source.func = smu.FUNC_DC_VOLTAGE
smu.source.offmode = smu.OFFMODE_HIGHZ
smu.source.level = batterycutoffv
smu.measure.terminals = smu.TERMINALS_FRONT
smu.source.range = batterycutoffv
smu.source.readback = smu.ON
smu.source.ilimit.level = currlimit
--Set the voltage limit for the battery to stop discharging
--Set the variable for number of iterations
voltLimit = batterycutoffv + 0.0001
iteration = 1
--Turn on the source output
smu.source.output = smu.ON
--Change display to user screen
display.changescreen(display.SCREEN_USER_SWIPE)
--Keep taking readings in the while loop until the measured voltage
--is equal to the voltage limit
while true do
   --Take a reading and get the current, voltage and relative timestamp
   curr = smu.measure.read(defbuffer1)

   volt = defbuffer1.sourcevalues[iteration]
   time = defbuffer1.relativetimestamps[iteration]
   hours = time/3600

   --Compare the measured voltage to the voltage limit
   --Exit the loop if it is
   if volt <= voltLimit then
      break
   end
   --Print the # of completed cycles, the voltage and the time for
   --the iteration. Display information on front panel
   print("Completed Cycles: ",iteration, "Voltage: ", volt, "Time: ", time)
   display.settext(display.TEXT1, string.format("Voltage = %.4fV", volt))
   display.settext(display.TEXT2, string.format("Current = %.2fA, Time = %.2fHrs", curr, hours))
   --Increment the number of iterations and wait 10 seconds
   iteration = iteration + 1
   delay(2.0)
end
--Turn the output off when the voltage limit is reached
smu.source.output = smu.OFF
FileNumber = file.open("/usb1/TestData.csv", file.MODE_WRITE)
file.write(FileNumber,"Current,Voltage,Seconds\n")
--Print out the measured values in a 4-column format
print("nIteration:\tCurrent:\tVoltage:\tTime:\n")
for i = 1, defbuffer1.n do
   print(i, defbuffer1[i], defbuffer1.sourcevalues[i], defbuffer1.relativetimestamps[i])
   file.write(FileNumber, string.format("%g,%g, %g\r\n",defbuffer1.readings[i], defbuffer1.
   sourcevalues[i],defbuffer1.relativetimestamps[i]))
end
file.close(FileNumber)
```

Appendix C: TSP Code Used to Apply a Continuous Cycling to an 18650 Lithium-Ion Battery

```
-- This code cycles a 3400 mAh 18650 Lithium-Ion battery. First the
-- battery is charged to its full potential using a hybrid constant
-- current, constant voltage method. The initial charge rate is 0.2C
-- (for 680 mA) which is applied until the battery reaches the threshold
-- voltage. Upon reaching this target, the source mode is changed and
-- 4.2 V is applied with a 680 mA limit. The source output remains on
-- until the measured current drops below 0.05C (170 mA).
-- Next, the battery is discharged to the cutoff voltage level using
-- a constant current method at a 0.2C rate (for -680 mA). Time
-- is capture through the discharge and capacity is computed at
-- the end of the cycle and displayed on the instrument front panel.
-- Cycling will continue until either the target cycle count is
-- reached or until the batter capacity falls below 80% of the
-- expected capacity.
reset()
local capacity = 3.400  -- in Ah
local crate = 0.2
local currlevel = capacity * crate
local thresholdv = 4.2
local nominalv = 3.6
local dischargecutoffv = 2.5
local deltaV = thresholdv - dischargecutoffv
local g_cyc_cnt = 0
local line_2_txt_1 = ""
local line_2_txt_2 = "Capacity: TBD Ah"
function configure_charge_and_discharge_settings()
   -- Start by defining the charge cycle
   -- Establish settings and configuration lists applicable to
   -- CC and CV modes of operation.
   -- Start with the CC source configuration
   smu.source.func = smu.FUNC_DC_CURRENT
   smu.source.range = currlevel     -- will apply appropriate source range
   smu.source.vlimit.level = thresholdv
   smu.source.offmode = smu.OFFMODE_HIGHZ
   smu.source.level = currlevel
   smu.source.configlist.create("sourcelist1")
   smu.source.configlist.store("sourcelist1", 1)

   -- Next add the measure configuration
   smu.measure.func = smu.FUNC_DC_VOLTAGE
   smu.measure.sense = smu.SENSE_4WIRE
   smu.measure.range = thresholdv      -- will apply appropriate measure range
   smu.measure.configlist.create("measurelist1")
   smu.measure.configlist.store("measurelist1", 1)

   -- Follow up with the CV source configuration
   smu.source.func = smu.FUNC_DC_VOLTAGE
   smu.source.offmode = smu.OFFMODE_HIGHZ
   smu.source.range = thresholdv
   smu.source.ilimit.level = currlevel
   smu.source.level = thresholdv
   smu.source.configlist.store("sourcelist1", 2)

   -- Then add the complementary measure configuration
   smu.measure.func = smu.FUNC_DC_CURRENT
   smu.measure.sense = smu.SENSE_4WIRE
   smu.measure.range = currlevel
   smu.measure.configlist.store("measurelist1", 2)

   -- Then define the discharge cycle.
   -- Because we want to do a CC discharge, we can reuse
   -- some of the charge cycle configuration.
   -- Recall the CC source/measure configuration
   smu.source.configlist.recall("sourcelist1", 1)
   smu.measure.configlist.recall("measurelist1", 1)

   -- Major change here is the current polarity.
   smu.source.level = currlevel * -1.0
```

```
   -- Store the configuration list settings. While the
   -- measure settings have not been altered, we still
   -- save this as a complementary index to make the
   -- logic a bit more understandable when we are
   -- performing settings recalls.
   smu.source.configlist.store("sourcelist1", 3)
   smu.measure.configlist.store("measurelist1", 3)
end
function do_charge_cycle()
   local txt_toggle = 0

   -- compute the delatV based on the upper threshold voltage
   deltaV = thresholdv - dischargecutoffv

   -- Recall the CC source/measure configuration
   smu.source.configlist.recall("sourcelist1", 1)
   smu.measure.configlist.recall("measurelist1", 1)

   display.clear()
   display.changescreen(display.SCREEN_USER_SWIPE)
   line_1_txt = string.format("Battery Cycle: %d", g_cyc_cnt)
   display.settext(display.TEXT1, line_1_txt)
   display.settext(display.TEXT2, "CC Charge Phase in Progress")

   -- Turn the output of the SMU on
   smu.source.output = smu.ON

   -- Loop voltage measurements at 2 s intervals until the
   -- threshold voltage is reached (+/-0.5%)
   local samplev = smu.measure.read()
   local stoppoint = thresholdv - (thresholdv * 0.005)
   local soc = 0
   while samplev < stoppoint do
      delay(2.0)  -- measure every 2 s
      samplev = smu.measure.read()
      if txt_toggle == 0 then
         display.settext(display.TEXT2, "CC Charge Phase in Progress")
         txt_toggle = 1
      else
         -- calculate state of charge
         soc = ((samplev - dischargecutoffv) / deltaV) * 100
         display.settext(display.TEXT2, string.format("SoC: %.2f %s", soc, string.char(37)))
         txt_toggle = 0
      end
   end

   -- Once the threshold is achieved, change to CV sourcing
   smu.source.configlist.recall("sourcelist1", 2)
   smu.measure.configlist.recall("measurelist1", 2)

   display.settext(display.TEXT2, "CV Charge Phase in Progress")

   smu.source.output = smu.ON -- is this even necessary???

   -- Now loop until the current is <= 0.05C
   local cutoffcurrrate = 0.05
   local cvcutoffcurr = capacity * cutoffcurrrate
   local samplei = smu.measure.read()
   while samplei > cvcutoffcurr do
      delay(2.0)  -- measure every 2 s
      samplei = smu.measure.read()
      samplev = defbuffer1.sourcevalues[defbuffer1.n]
      if txt_toggle == 0 then
         display.settext(display.TEXT2, "CV Charge Phase in Progress")
         txt_toggle = 1
      else
         -- calculate state of charge
         soc = ((samplev - dischargecutoffv) / deltaV) * 100
         display.settext(display.TEXT2, string.format("SoC: %.2f %s", soc, string.char(37)))
         txt_toggle = 0
      end
```

```
   end
end
function do_discharge_cycle()
   local txt_toggle = 2

   -- Recall the CC source/measure configuration
   smu.source.configlist.recall("sourcelist1", 3)
   smu.measure.configlist.recall("measurelist1", 3)

   display.clear()
   display.changescreen(display.SCREEN_USER_SWIPE)
   line_1_txt = string.format("Battery Cycle: %d", g_cyc_cnt)
   display.settext(display.TEXT1, line_1_txt)
   line_2_txt_1 = "CC Discharge Phase in Progress"
   display.settext(display.TEXT2, line_2_txt_1)
   -- Turn the output of the SMU on
   smu.source.output = smu.ON

   timer.cleartime()

   -- Loop voltage measurements at 2 s intervals until the
   -- discharge cutoff voltage is reached.
   local samplev = smu.measure.read()
   local l_capacity = 0
   local dod = 0

   -- compute the delatV based on the upper nominal voltage
   deltaV = thresholdv - dischargecutoffv

   -- add in a means to compute capacity and return this
   while samplev > dischargecutoffv do
      delay(2.0)  -- measure every 2 s
      samplev = smu.measure.read()
      elapsed_t = timer.gettime()-- capture elapsed relative time in seconds
      if txt_toggle == 2 then
         l_capacity = math.abs(smu.source.level) * (elapsed_t / 3600)
         line_2_txt_2 = string.format("Capacity: %.4f Ah", l_capacity)
         display.settext(display.TEXT2, line_2_txt_2)
         txt_toggle = 1
      elseif txt_toggle == 1 then
         display.settext(display.TEXT2, line_2_txt_1)
         txt_toggle = 0
      elseif txt_toggle == 0 then
         -- calculate state of charge
         dod = (1 - (samplev - dischargecutoffv) / deltaV) * 100
         display.settext(display.TEXT2, string.format("DoD: %.2f %s", dod, string.char(37)))
         txt_toggle = 2
      end
   end

   -- compute capacity and return
   l_capacity = math.abs(smu.source.level) * (elapsed_t / 3600)
   line_2_txt_2 = string.format("Capacity: %.4f Ah", l_capacity)
   display.settext(display.TEXT2, line_2_txt_2)
   return l_capacity
end
function do_cycling(cycle_count, charge_rest_interval_sec, discharge_rest_interval_sec, tol_capacity)
   local meas_capacity = 0

   while g_cyc_cnt < cycle_count do
      g_cyc_cnt = g_cyc_cnt + 1
      do_charge_cycle()

      if charge_rest_interval_sec > 0.0 then
         -- Turn the output of the SMU off
         smu.source.output = smu.OFF
         delay(charge_rest_interval_sec)
      end


      meas_capacity = do_discharge_cycle()
      if meas_capacity < (tol_capacity/1000) then
```

```
         break
      else
         if discharge_rest_interval_sec > 0.0 then
            -- Turn the output of the SMU off
            smu.source.output = smu.OFF
            delay(discharge_rest_interval_sec)
         end
      end
   end

   -- Turn the output of the SMU off
   smu.source.output = smu.OFF
end
local target_cycle_count = 500
local chrg_rest_sec = 10.0
local dischg_rest_sec = 10.0
local cutoffcapacity = 3400 * 0.8
configure_charge_and_discharge_settings()
do_cycling(target_cycle_count, chrg_rest_sec, dischg_rest_sec, cutoffcapacity)
```

Contact Information:

Australia 1 800 709 465
Austria* 00800 2255 4835
Balkans, Israel, South Africa and other ISE Countries +41 52 675 3777
Belgium* 00800 2255 4835
Brazil +55 (11) 3530-8901
Canada 1 800 833 9200
Central East Europe / Baltics +41 52 675 3777
Central Europe / Greece +41 52 675 3777
Denmark +45 80 88 1401
Finland +41 52 675 3777
France* 00800 2255 4835
Germany* 00800 2255 4835
Hong Kong 400 820 5835
India 000 800 650 1835
Indonesia 007 803 601 5249
Italy 00800 2255 4835
Japan 81 (3) 6714 3086
Luxembourg +41 52 675 3777
Malaysia 1 800 22 55835
Mexico, Central/South America and Caribbean 52 (55) 88 69 35 25
Middle East, Asia, and North Africa +41 52 675 3777
The Netherlands* 00800 2255 4835
New Zealand 0800 800 238
Norway 800 16098
People's Republic of China 400 820 5835
Philippines 1 800 1601 0077
Poland +41 52 675 3777
Portugal 80 08 12370
Republic of Korea +82 2 565 1455
Russia / CIS +7 (495) 6647564
Singapore 800 6011 473
South Africa +41 52 675 3777
Spain* 00800 2255 4835
Sweden* 00800 2255 4835
Switzerland* 00800 2255 4835
Taiwan 886 (2) 2656 6688
Thailand 1 800 011 931
United Kingdom / Ireland* 00800 2255 4835
USA 1 800 833 9200
Vietnam 12060128
*** European toll-free number. If not accessible, call:** +41 52 675 3777

Rev. 02.2022

KEITHLEY
A Tektronix Company

Find more valuable resources at TEK.COM

121622 SBG 1KW-73979-0