Getting started with AutoDevKit battery management system (BMS) evaluation boards

Introduction

In a multicell battery pack, placing cells in series augments the possibilities of cell imbalance, which equates to a slower but persistent degradation of the battery.

There are always slight differences in the state of charge (SOC), self-discharge rate, capacity, impedance, and temperature characteristics, even for cells of the same model from the same manufacturer and even from the same batch of production.

These differences can lead to a divergence in a battery cell voltage over time. Cells with lower capacity or higher internal impedance tend to have higher voltage than the rest of the series cells at full charge. These cells are weakened further by continuous overcharge cycles. The higher voltage of weak cells at charge completion causes accelerated capacity degradation.

On the other hand, in discharge, the weak cells tend to have lower voltage than the other cells, due to either higher internal resistance or the faster rate of discharge that results from their smaller capacity. This means that if any of the weak cells hits the cell undervoltage-protection limit while the pack voltage is still sufficient to power the system, the full capacity of the battery is not used.

One of the emerging technologies for enhancing battery safety and extending battery life is advanced cell balancing. Since new cell balancing technologies track the amount of balancing needed by individual cells, the usable life of battery packs is increased, and overall battery safety is enhanced.

A battery management system (BMS) can manage battery cells, enhancing their safety and duration.

A BMS combines hardware and software to monitor and send acquired data to a dedicated device to protect the battery from overload or over discharge, lengthening its life cycle.

In the earlier BMSs, the SOC estimation was based only on the voltage reading.

Nowadays, instead, this estimation is based on Kalman filters, which provide more accurate data on the SOC by reading the voltage, the current, and the temperature of battery cells.

The modern BMS includes functions such as cell monitoring, balancing, safety and protection of the batteries, the state of charge estimation and thermal management:

- Cell monitoring: this function is based on the acquisition of the current, the voltage, and the temperature of each cell of the battery pack.
- Cell balancing: consists of distributing the energy and maintaining the SOC of the cells at a similar level. There are two different types of balancing: active and passive. Active balancing consists of transferring the exceeding energy of cells with a higher SOC to cells with a lower SOC to reach the same level. Passive balancing dissipates the exceeding energy of cells with a higher SOC, generating heat. Cell balancing is a key function to maintain the battery capacity and lengthen its duration.
- Battery safety and protection: a BMS ensures that the battery works in safe conditions for the users and the battery itself.
- State of charge (SOC) and state of health (SOH): an accurate SOC estimation is necessary to lengthen the battery life cycle, prevent battery damage, and ensure efficiency and accurate calculations of the SOH and cell balancing.

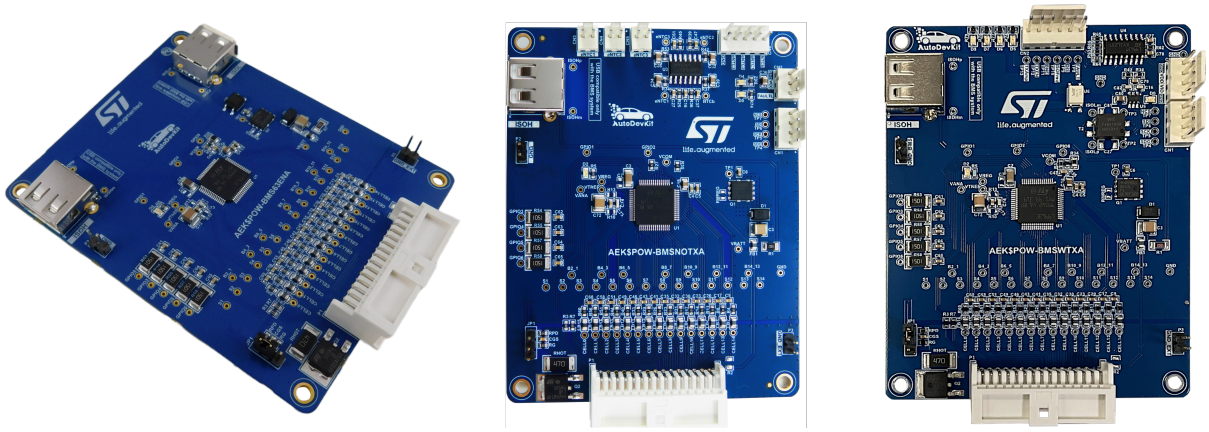
We offer a variety of BMS evaluation boards (AEK-POW-BMS63EN, AEK-POW-BMSNOTX, AEK-POW-BMSWTX) to meet various customers' needs and help designers build different types of BMS chain topologies fit for any end-user application.

These boards are included in the AutoDevKit ecosystem to make design and prototyping even faster and easier. Next chapters focus on a comprehensive and detailed description of all the boards belonging to our BMS range and the various topologies of BMS chains you can experimenting with.

Note: The evaluation boards described in this user manual are designed for R&D laboratory use only. They are not intended for field use in vehicles.

Moreover, they are not reference designs. Their purpose is evaluation and not production as stated in our Terms of use.

Figure 1. AutoDevKit BMS evaluation board range



Note: For dedicated assistance, please submit a request in our AutoDevKit Community.

1 BMS evaluation board overview

1.1 AEK-POW-BMS63EN overview

Our AEK-POW-BMS63EN is a battery management system (BMS) evaluation board that can handle from 1 to 31 Li-ion battery nodes. Each battery node manages from 4 to 14 battery cells, for a voltage range between 48 V and 800 V.

The board is based on the L9963E, which is designed for operation in both hybrid (HE) and full electric (BE) vehicles using lithium battery packs, but its use can be extended to other Transportation and Industrial applications.

The L9963E is an ASIL-D, AEC-Q1000 qualified device, compliant with ISO26262.

The main activity of the L9963E is monitoring cells and battery node status through stack voltage measurement, cell voltage measurement, temperature measurement, and coulomb counting. Measurement and diagnostic tasks can be executed either on demand or periodically, with a programmable cycle interval. Measurement data are available for an external microcontroller to perform charge balancing and to compute the state of charge (SOC) and the state of health (SOH).

The AEK-POW-BMS63EN can measure 5 values of temperature through 5 GPIOs (which can be configured as analog inputs to connect NTCs).

The AEK-POW-BMS63EN provides an elaborate monitoring network to sense the voltage, current, and temperature of each cell.

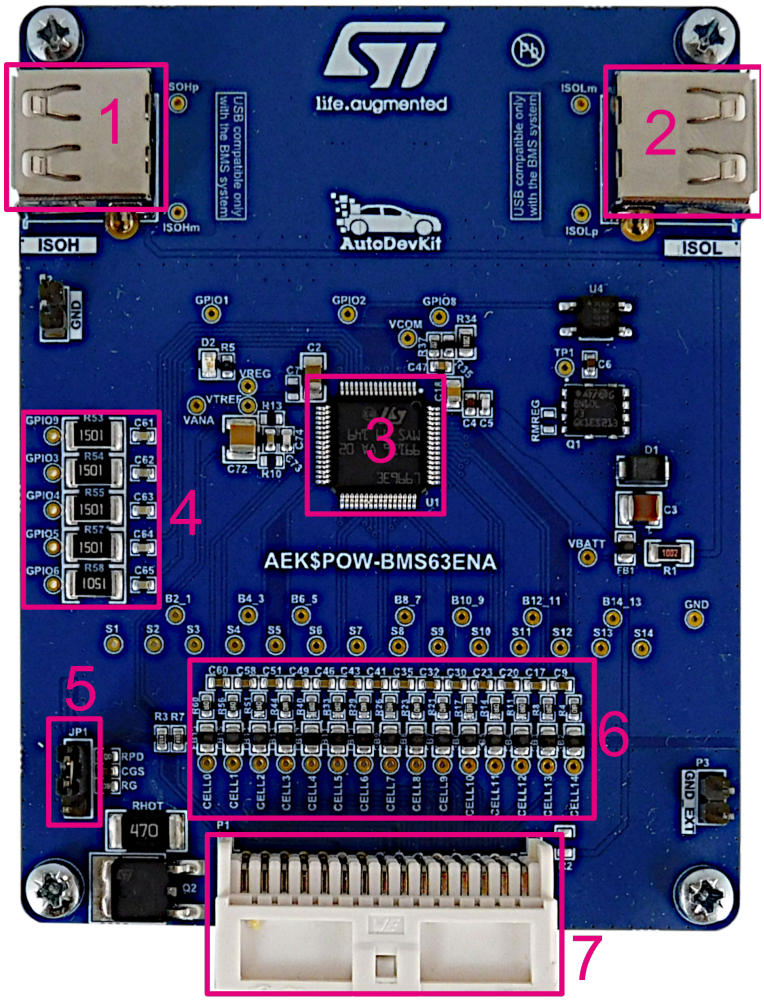
1.1.1 Features

- Hosts the L9963E AEC-Q100 qualified automotive multicell battery monitoring and balancing IC
- Voltage monitoring of every single cell and of the entire battery node
- Voltage, current, and temperature sensing of each cell
- 5 GPIOs to connect temperature sensors as NTCs
- An NTC hosted on the L9963E to sense the chip temperature
- Passive balancing
- Compact size: 100 mm x 76 mm
- Included in the AutoDevKit ecosystem

1.1.2 Main components

1. ISOH port to connect the board to another AEK-POW-BMS63EN in a daisy chain
2. ISOL port to connect the board to the AEK-COM-ISOSPI1
3. L9963E
4. GPIOs for external NTC connection
5. Hot plug protection
6. Balancing resistors
7. Connector for the battery pack

Figure 2. **AEK-POW-BMS63EN main components**



1.2 AEK-POW-BMSNOTX overview

Our offer for BMS evaluation boards also includes the AEK-POW-BMSNOTX that manages from 4 to 14 battery cells.

Like the AEK-POW-BMS63EN, this board is based on the L9963E.

This embedded L9963E device can act as a transceiver, directly communicating with an MCU via SPI.

The board is particularly fit for auxiliary battery systems to supply power for devices (such as audio system, window cleaning

system, seat heating, light system, light signalization, climate control system) connected to your vehicle (even when the engine

is not running), ensuring the main starting battery is reserved for engine cranking and vehicle electrical requirements.

1.2.1 Features

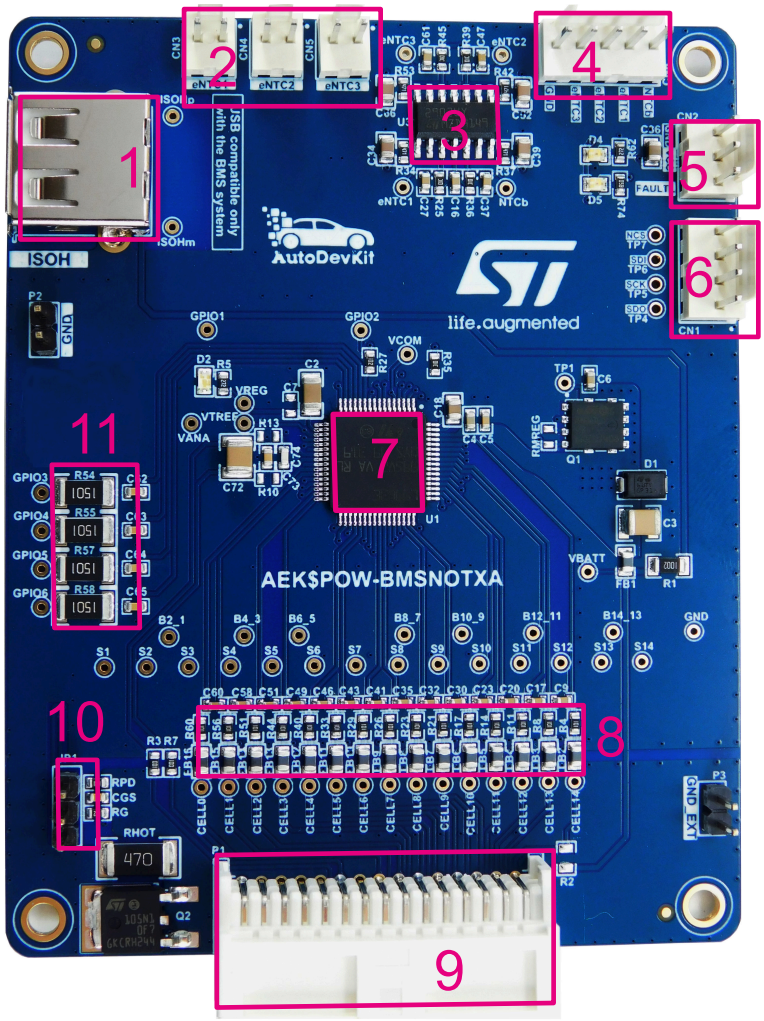
- The board hosts the L9963E AEC-Q100 qualified automotive multicell battery monitoring and balancing IC
- Voltage monitoring of every single cell and of the entire battery node
- Current sensing of the entire battery node
- 4 GPIOs to connect NTC sensors present on the battery pack connector
- 3 NTC sensors that can be connected to CN3, CN4 and CN5 connectors
- 1 NTC sensor to measure the board temperature
- CN1 connector to communicate with an MCU board via SPI

- CN2 connector for diagnostic functions
- CN6 connector for MCU ADCs dedicated to the NTC sensors reading
- Passive balancing available
- Compact size: 100 mm x 76 mm
- Included in the AutoDevKit ecosystem

1.2.2 Main components

1. ISOH port to connect the board to an AEK-POW-BMS63EN in a daisy chain
2. CN3, CN4 and CN5 connectors to connect three external NTC sensors
3. LM2902W Low power quad operational amplifier
4. CN6 connector for MCU ADCs dedicated to the NTC sensors reading
5. CN2 connector for diagnostic functions
6. CN1 connector to communicate with an MCU board via SPI
7. L9963E AEC-Q100 qualified automotive multicell battery monitoring and balancing IC
8. Balancing resistors
9. Connector for the battery pack
10. Hot plug protection
11. GPIOs for external NTC connection handled by L9963E

Figure 3. AEK-POW-BMSNOTX main components



1.3 AEK-POW-BMSWTX overview

The AEK-POW-BMSWTX is a battery management system (BMS) evaluation board that manages from 4 to 14 battery cells.

The main advantage of this evaluation board is ensuring isolated connection to an external MCU, thanks to the embedded transceiver.

The board is based on the L9963E Li-ion battery monitoring and protection chip for high-reliability automotive applications and the L9963T general purpose SPI to isolated SPI bidirectional transceiver.

The main activity of the L9963E is monitoring the cells and battery node status through stack voltage measurement, cell voltage measurement, temperature measurement, and coulomb counting.

Measurement and diagnostic tasks can be executed either on demand or periodically, with a programmable cycle interval.

Measurement data are available for an external microcontroller to perform charge balancing and to compute the state of charge (SOC) and the state of health (SOH).

The L9963T general purpose SPI to isolated SPI bidirectional transceiver can transfer communication data incoming from a classical 4-wire based SPI interface to a 2-wire isolated interface (and vice versa). In our board, the transceiver is configured as a slave.

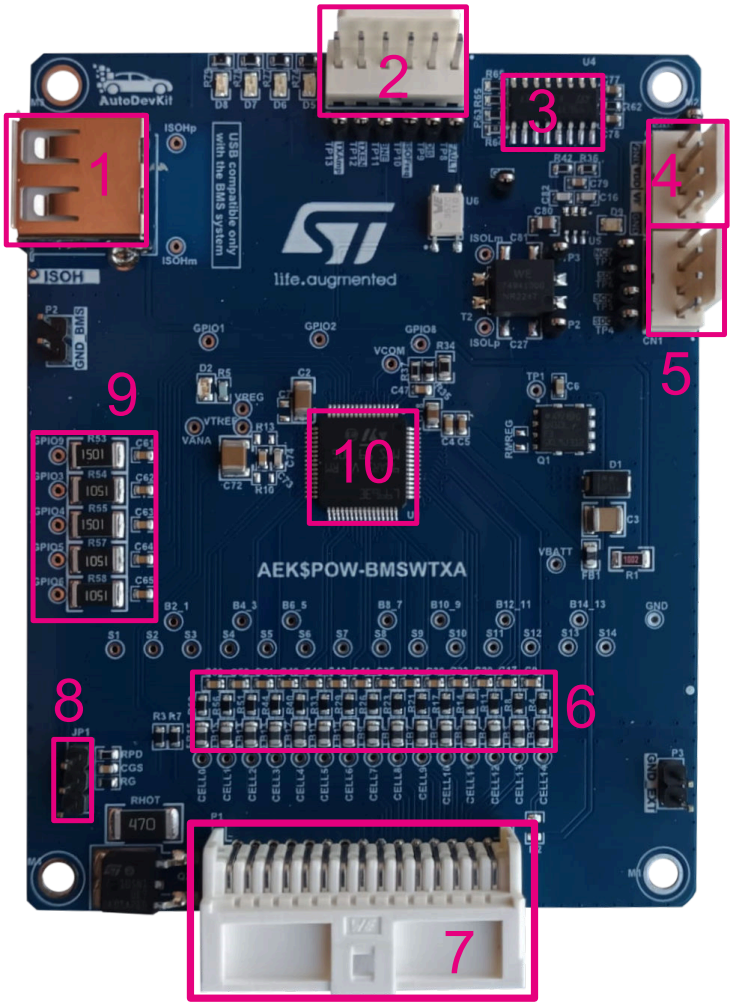
1.3.1 Features

- Hosts the L9963E AEC-Q100 qualified automotive multicell battery monitoring and balancing IC
- Hosts the L9963T AEC-Q100 qualified automotive general purpose SPI to isolated SPI bidirectional transceiver
- Voltage monitoring of every single cell and of the entire battery node
- Current sensing of the entire battery node
- 5 GPIOs to connect temperature sensors as NTCs
- CN1 connector that allows establishing communication with an MCU board via SPI
- CN2 connector that interfaces directly to an MCU board for control and diagnostic functions
- Passive balancing
- Compact size: 100 mm x 76 mm
- Included in the AutoDevKit ecosystem

1.3.2 Main components

1. ISOH port to connect the board to an AEK-POW-BMS63EN in a daisy chain
2. Connector for MCU ADCs dedicated to the NTC sensors reading
3. Automotive general-purpose SPI to isolated SPI transceiver
4. CN2 connector for diagnostic functions
5. CN1 connector to communicate with an MCU board via SPI
6. Balancing resistors
7. Connector for the battery pack
8. Hot plug protection
9. GPIOs for external NTC connection handled by L9963E
10. L9963E AEC-Q100 qualified automotive multicell battery monitoring and balancing IC

Figure 4. **AEK-POW-BMSWTX main components**



1.4 Embedded devices

1.4.1 L9963E

The L9963E is intended for operation in both hybrid electric (HE) and full electric (FE) vehicles using lithium battery packs. The IC embeds all the features needed to perform battery management. A single device can monitor from 4 up to 14 cells.

The device can be supplied with the same battery it monitors.

The L9963E main activity consists of monitoring cells and battery pack status through stack voltage measurement, cell voltage measurement, temperature measurement, and coulomb counting. Measurement and diagnostic tasks can be executed either on demand or periodically, with a programmable cycle interval.

Measurement data is available for an external microcontroller to perform charge balancing and to compute the state of health (SOH) and state of charge (SOC).

The IC works in normal mode performing measurement conversions, diagnostics, and communication. The device can also be put into a cyclic wakeup state in order to reduce the current consumption from the battery.

Passive cell balancing can be performed either via internal discharge path or via external MOSFETs. The controller can either manually control the balancing drivers or start a balancing task with a fixed duration. In the second case, the balancing may be programmed to continue also when the IC enters a low power mode called silent balancing, to avoid unnecessary current absorption from the battery pack.

Thanks to the GPIOs, the device also offers the possibility to operate a distributed cell temperature sensing via external NTCs resistances.

The external microcontroller can communicate with L9963E via SPI protocol. The physical layer can either be a classic 4-wire based SPI or 2-wire transformer/capacitive based isolated interface through a dedicated isolated transceiver device.

The L9963E performs automatic validation of any failure involving the cells or the whole battery pack. The device can detect the loss of the connection to a cell or GPIO terminal. Moreover, it features a hardware self-check (HWSC) that verifies the correct functionality of the internal analog comparators and the ADCs. All these checks are automatically performed in case a failure involving both cells or when the battery pack is detected. The current sensing interface used for coulomb counting is also capable of detecting failures such as open wires and overcurrent in sleep mode. The cell balancing terminals can detect any short/open fault and the internal power MOS are protected against overcurrent.

1.4.2 L9963T

L9963T is a general purpose SPI to isolated SPI transceiver intended to create a communication bridge between devices located into different voltage domains.

L9963T is able to transfer communication data incoming from a classical 4-wire based SPI interface to a 2-wire isolated interface (and viceversa).

The transceiver supports both transformer and capacitive isolation, since the isolated signal generated according to a proprietary protocol is suitable to be transmitted over both decoupling circuitries.

The device can be configured either as Slave or as Master of the SPI bus and supports any protocol made of SPI frames 8 to 64 bit long. The transceiver manages the transfer of the information without performing any protocol check.

SPI peripheral can work up to 10 MHz when configured as Slave. SPI clock frequency can be programmed among (250 kHz; 1 MHz; 4 MHz; 8 MHz) when configured as Master.

Isolated SPI peripheral features two different operating modes: slow @333 kbps and fast @2.66 Mbps.

The asynchronicity between the two sides is internally managed, allowing all possible configuration frequencies on both peripherals to be used in application.

L9963T features an internal queue of 3 slots for the frames received on the SPI port and a queue of 20 slots for the ones received on the isolated SPI side. This allows buffering and decoupling the two different clock domains.

The device is natively compatible with L9963 isolated SPI, allowing its usage in the BMS applications.

L9963T is compatible with both 3.3 V and 5 V logics.9963T.

1.5 Voltage operating range

In our BMS evaluation boards, the maximum voltage range for each cell is 4.2 V.

The power supply range is from 9.6 V to a maximum of 64 V.

1.5.1 Linear regulators

Our BMS boards feature several linear voltage regulators, which are switched on according to a specific sequence at power-up (see Figure 12. Finite state machine of the voltage conversion routine).

VREG

This linear regulator exploits an external MOS to decrease the power dissipation inside the L9963E.

It acts as a pre-regulator, supplying all other internal regulators (VANA, VCOM, VTREF, and VDIG). It is switched off in low power modes (sleep, silent balancing, off phase of the cyclic wakeup).

VANA

This low drop regulator supplies all the ADC, comparators, monitors, main bandgap, current generator, and other analogic blocks.

VCOM

The isolated communication receiver/transmitter and the GPIO output buffers are supplied by this low drop regulator.

VTREF

This low drop regulator is used to supply external components such as NTCs for temperature sensing.

The recommended application circuit in NTC analog front end guarantees that each NTC channel sinks no more than 500 µA.

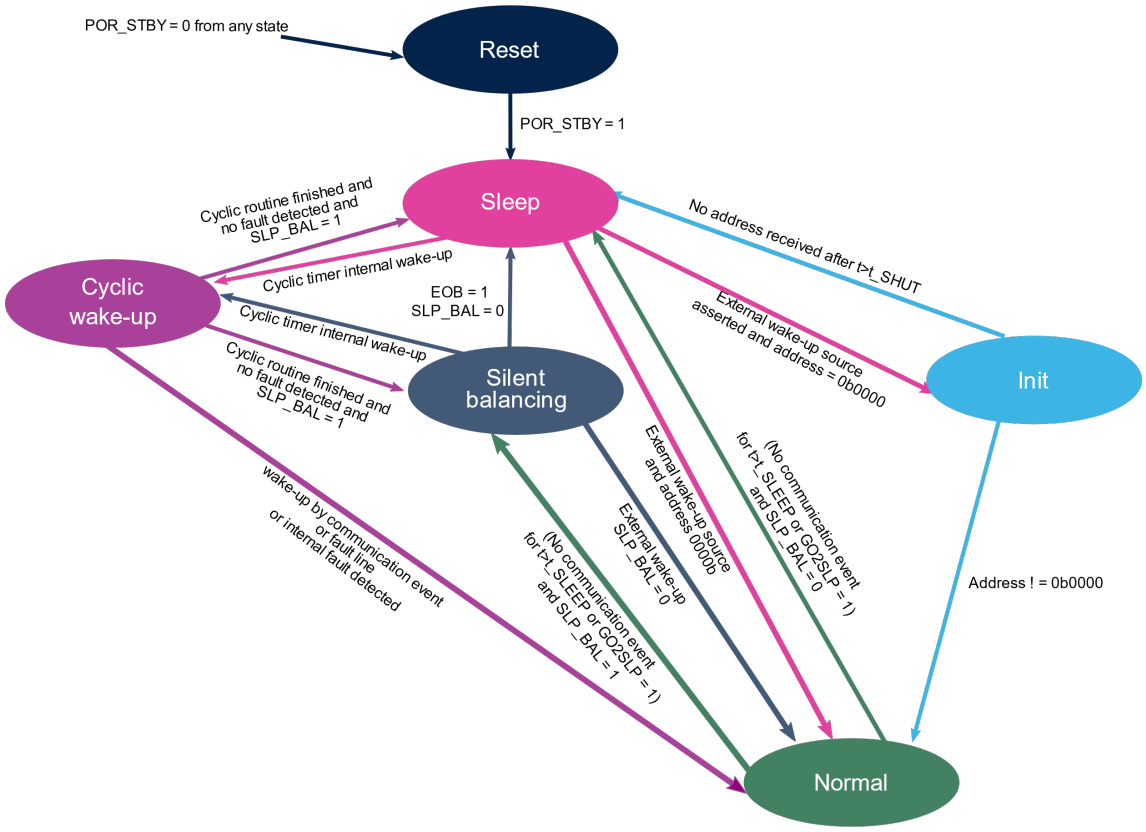
VTREF regulator is disabled by default. Its operation can be controlled via SPI.

In absolute measurements, there is no reference value, while the ratiometric measurement is based on reference value defined by the VTREF regulator. If the VTREF goes low in case of an error, the VTREF varies to compensate this error.

All of the above regulators have dedicated UV/OV diagnostics.

1.5.2 Simplified state machine

Figure 5. State machine



1.5.2.1 *Reset and sleep states*

When the standby logic is reset, all registers on the device are in the reset state. The battery voltage is still under threshold. No operation is possible during this state.

The sleep state is reached:

- From the reset state, when POR_STBY rises.
- From other states, if a GO2SLP command is sent or no communication is received for t > t_SLEEP.
- From the init state in case the device address is still 0b0000 after t > t_SHUT.
- From the cyclic wakeup state when the silent balancing is not resumed.

In the sleep state, the device is sensitive to external sources (such as ISO lines, fault line, SPI_CS (SPI_CLK) pins, and a GPIO pin for master units) to wake up the main logic. A slow oscillator works in this state to allow the device to wake up every t = t_{CYCLIC_SLEEP} + t_{CYCLIC_WUP} and move to the cyclic wakeup state.

During the sleep state, the current consumption is significantly reduced to the I_{SLEEP} current value: only the communication wakeup source monitoring, low-speed oscillator for cyclic wakeup timer, and the corresponding reference and power supply are activated.

1.5.2.2 *Init state*

In the init state, after having been woken up, the device waits for the microcontroller to send the address assignment command (refer to Section 1.5.2.2.1: Addressing procedure). If the address command is received before the initialization timer expires (t_SHUT), the device address is stored into a standby logic register (chip_ID) and the device enters the normal state.

The chip_ID field is then locked and no longer editable. Two actions can correctly re-initialize the device (including the chip_ID):

- Hardware reset: (POR_STBY)
- Software reset: set SW_RST and GO2SLP in the same frame

Note: the software reset leaves the communication timeout (CommTimeout) unmodified, and clears the chip_ID.

If only SW_RST is sent, the device waits for CommTimeout and then moves to the sleep state.

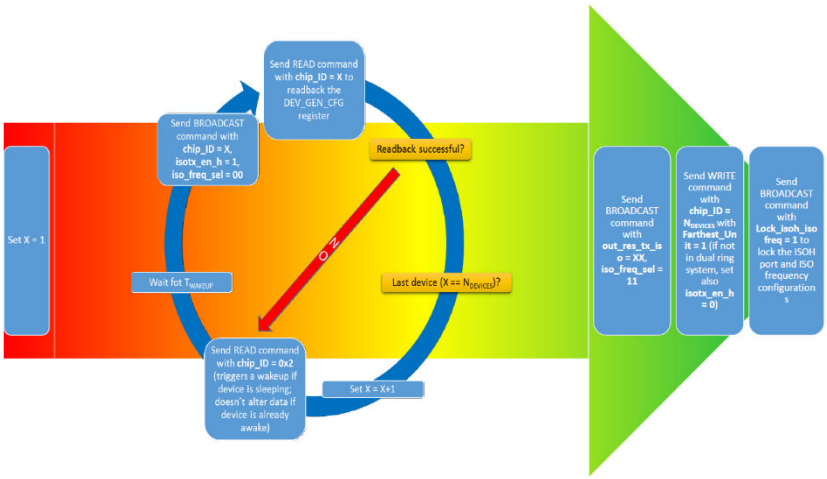
If the initialization timer (t_SHUT) expires before the command is received, the device goes back to the sleep state.

Any failure is masked until the device receives an address.

1.5.2.2.1 Addressing procedure

The following figure shows the daisy-chain addressing procedure for a stack of $N_{DEVICES}$.

Figure 6. Daisy chain addressing procedure

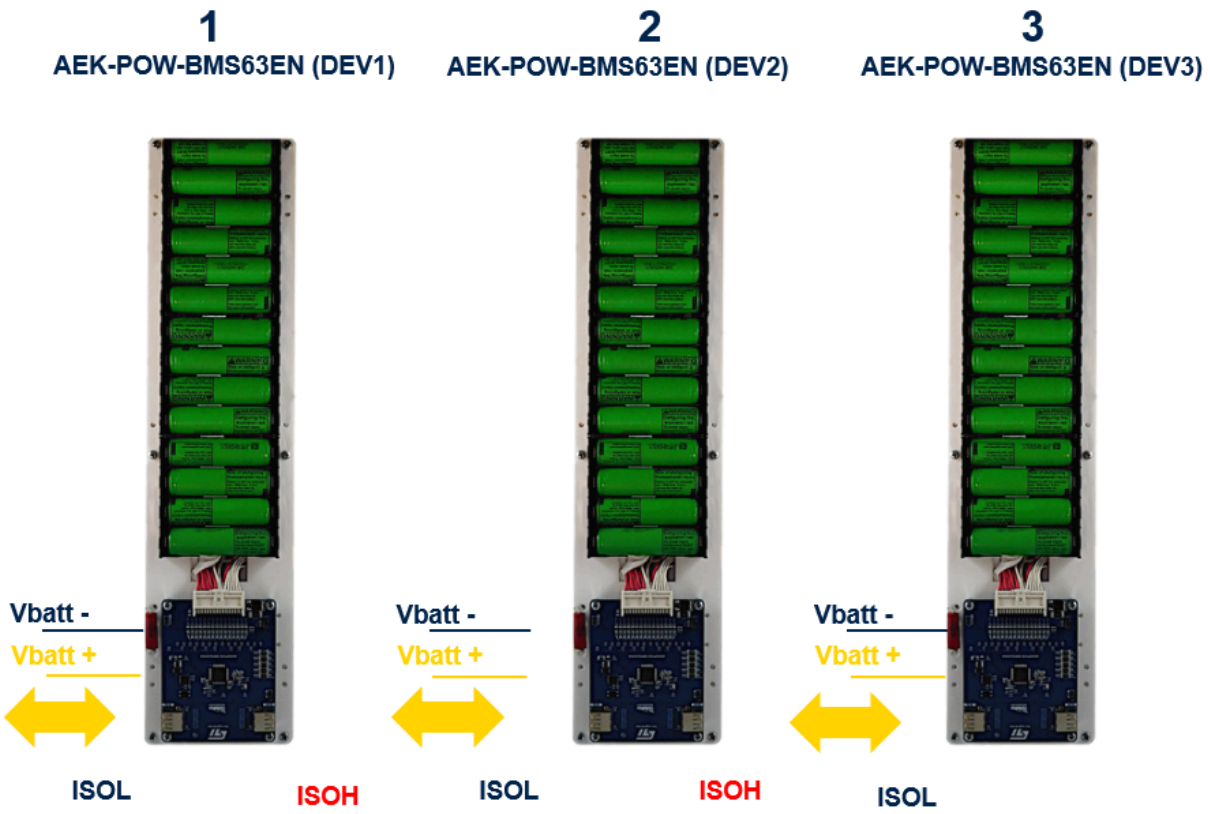


1.5.2.2.2 Addressing procedure

The addressing procedure assign a unique ID to each chain board.

IDs are sequentially assigned to all the BMS included in the system: ID no. 1 is assigned to the first BMS in the chain, ID no. 2 is assigned to the second, and so on.

Figure 7. **Addressing procedure example for the AEK-POW-BMS63EN**



To determine if the addressing procedure has been successful, check the D2 LED on the BMS. If it remains always on, it means that the procedure has been correctly executed and each BMS has its own ID. Only if you unplug the connector, the BMS loses their ID.

1.5.2.3 *Normal state*

In this state, all references are powered. ADCs and interfaces are ready for measurement and data transmission, respectively.

The commands sent by the microcontroller can be read from both ISO lines and SPI pins.

When receiving a valid command, the L9963E executes the corresponding operations, such as voltage, current, and overtemperature measurements.

Some safety operations (OV, UV, OT, and VBAT monitoring) are automatically checked in background.

If the communication with the MCU is missing for t > t_SLEEP (programmable via CommTimeout, maskable via comm_timeout_dis) or a GO2SLP command is received, the device moves either to the sleep state or to the silent balancing state, depending on the slp_bal_conf bit and balancing state.

1.5.2.4 *Silent balancing state*

In this state, one or more cells are balanced with a reduced current consumption with respect to the normal state.

Active resources are the same as the ones of the sleep state plus the balance drivers and the necessary bias circuitry.

To enter the silent balancing state from the normal state, check the following conditions:

1. Cell balancing must be on.
2. The slp_bal_conf flag has to be set to 1.
3. Verify the “go to sleep” condition (an explicit GO2SLP command or communication timeout expiration).

If cell balancing is previously requested in the normal state and the slp_bal_conf flag is set to 1, when a condition to go to sleep (low consumption) occurs, the device enters the silent balancing state.

To exit from the silent balancing state, there are three possible events:

- A wakeup signal on communication or fault line can force the chip to stop balancing and then go back to the normal state.
- An external fault must bring the device to the normal state and stop the balancing.
- As soon as the required balancing target stops, the end of balancing (EOB) bit is set to 1 and the chip enters the sleep state.

If the Cyclic signal is raised, the device goes to the cyclic wakeup state, runs the diagnosis, and then resumes silent balancing (if slp_bal_conf flag = 1).

1.5.2.5 *Cyclic wakeup state*

From both sleep and silent balancing states, the device moves periodically (once every t_{CYCLIC_SLEEP}) to the cyclic wakeup state to monitor eventual faults.

The ADC must be on to check possible critical battery conditions. Any detected fault moves the device to the normal state.

An on-demand operation is possible only once the device has moved to the normal state in case of any detected fault.

Conditions to leave this state are:

- Any fault detected moves the device to the normal state.
- A wakeup from the fault line or communication line moves the device to the normal state if the defined monitoring tasks are finished; the device can move to the sleep or silent balancing states automatically based on the state before cyclic conversions (slp_bal_conf flag).

2 BMS topologies

Our BMS boards can work in two different daisy chain topologies: centralized and dual access ring.

Note: To quickly create a battery pack for our BMS solutions, the AutoDevKit ecosystem has been extended to include a specific cylindrical battery holder (AEK-POW-BMSHOLD).

Note: The daisy chain and dual access ring topologies are based on a chain of BMS evaluation boards. The chain first node can be an AEK-POW-BMSWTX, an AEK-POW-BMSNOTX or an AEK-POW-BMSWTX. The remaining nodes must be all AEK-POW-BMS63EN evaluation boards.

2.1 Centralized configuration

In a centralized daisy chain configuration, a series of BMS is connected to an MCU board (e.g., AEK-MCU-C4MLIT1) through a single transceiver connected to the AEK-POW-BMS63EN isolated ISOL port. The BMS are connected to each other through the isolated ISOH port.

The MCU communicates with the AEK-COM-ISOSPI1 hosted L9963T transceiver through the SPI protocol. The transceiver converts these signals into ISO SPI signals to communicate with the BMS.

Figure 8. Centralized BMS diagram

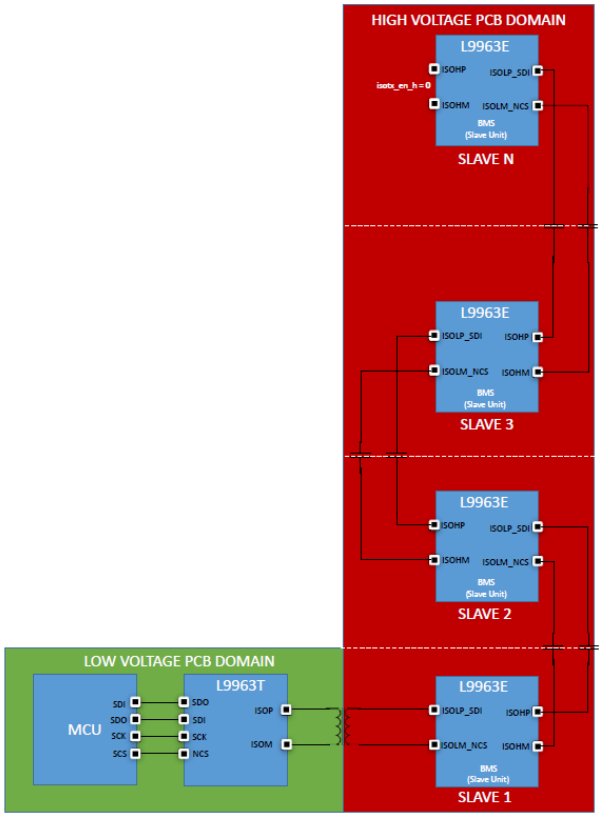
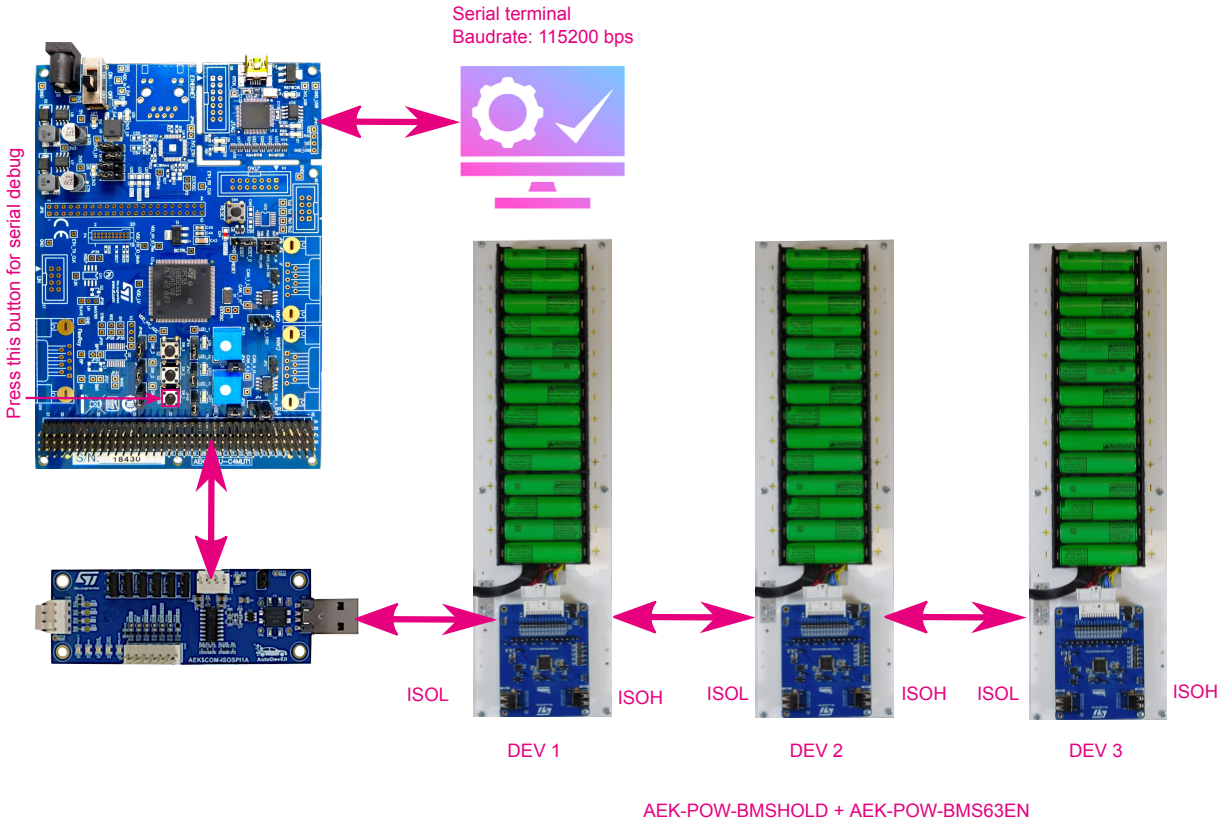


Figure 9. **Example of centralized BMS chain using the AEK-POW-BMS63EN with the AEK-COM-ISOSPI1 and the AEK-MCU-C4MLIT1**



2.2 Dual access ring configuration

A dual access ring configuration is realized by adding another transceiver that makes the communication bidirectional. The secondary ring is used as a backup in case the primary ring fails. Data moves in opposite directions around the rings, and each ring remains independent of the other unless the primary ring fails. The two rings are connected to continue the flow of data traffic.

Figure 10. **Dual access ring BMS diagram**

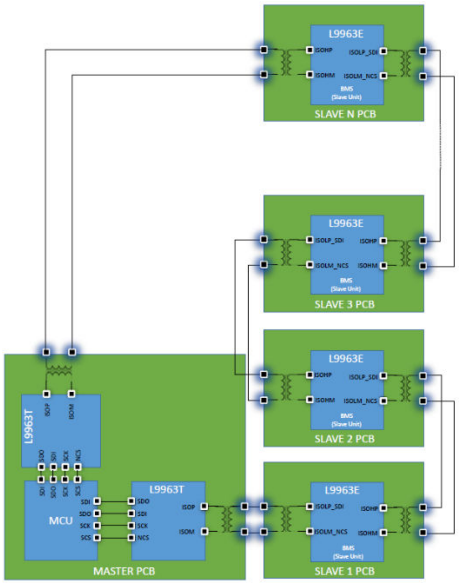
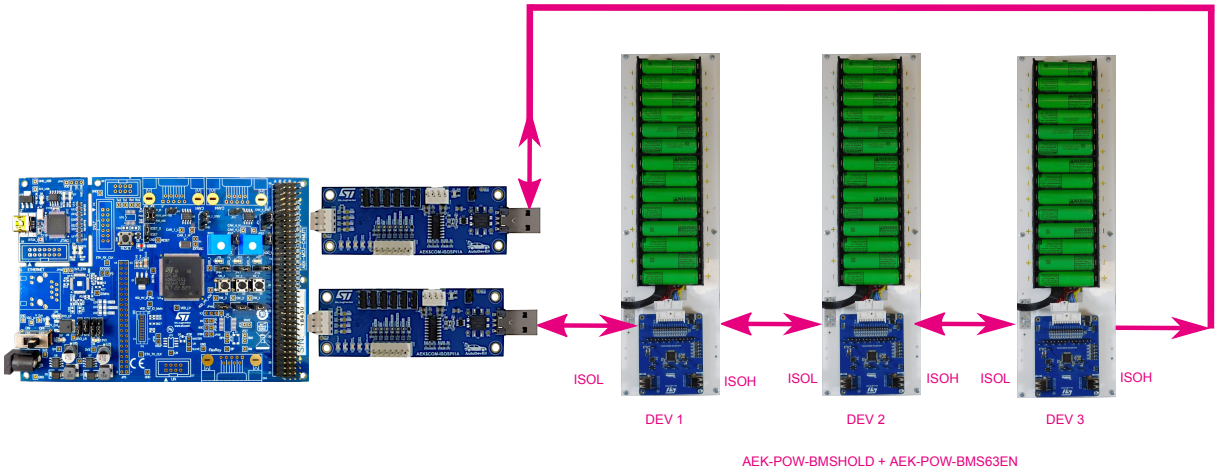


Figure 11. **Dual ring configuration example for the AEK-POW-BMS63EN**



3 Voltage conversion routine

L9963E implements a flexible voltage conversion routine, whose main goals are:

- Providing on-demand information about the cell voltage, the stack voltage, and the cell temperature.
- Providing on-demand diagnostic information about device functionality.
- Periodically monitoring the cells and the stack status, along with the device functionality.
- Limit power consumption by activating only the necessary resources.
- Automatically validate any eventual failure detected during the routine execution.

Through the voltage conversion routine we can also read voltages from the GPIOs configured as analog inputs. Thus, connecting NTCs to GPIOs, we can measure the battery pack temperature.

The voltage conversion routine can be executed in three different ways according to the microcontroller commands. The different modes are mutually exclusive: only one routine at a given time is allowed and multiple threads are not supported.

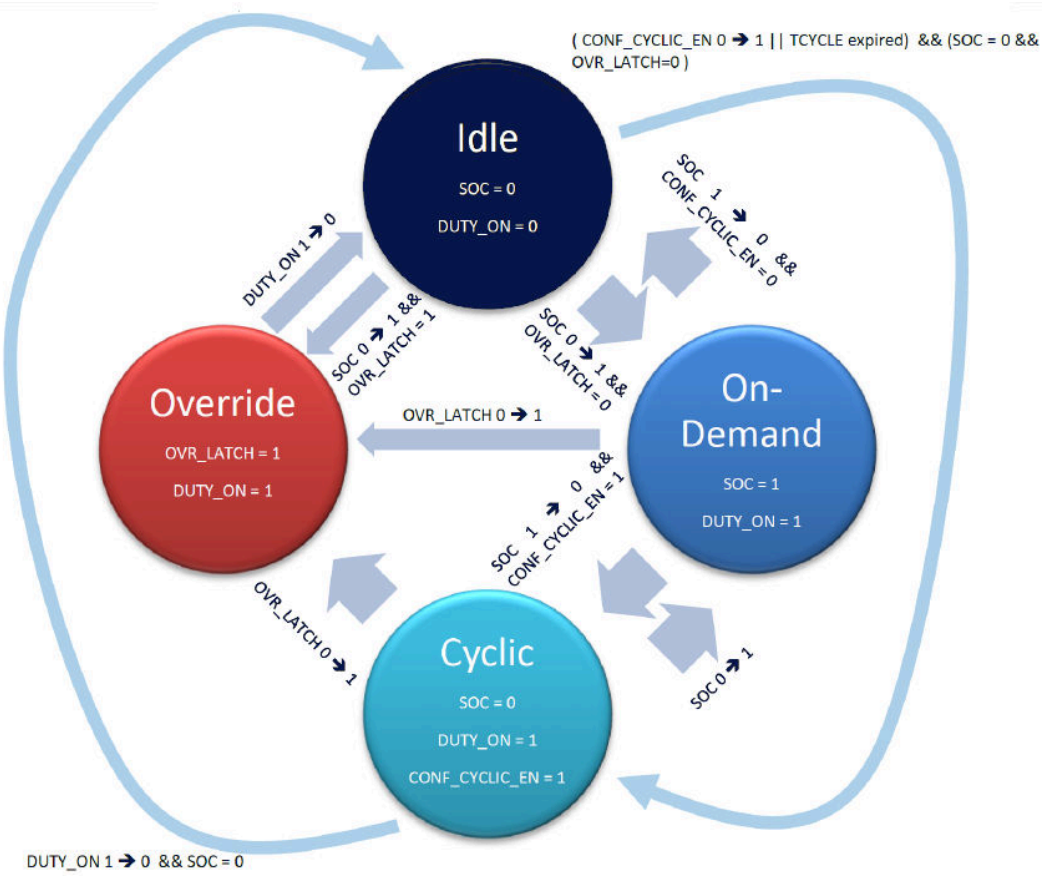
The execution modes have different priorities:

- Configuration override is high priority since its purpose is to perform diagnostics upon failure detection to validate the failure type. It can interrupt any ongoing activity and, once done, the voltage conversion routine is moved to the idle state, waiting for the microcontroller to interpret the diagnostic data.
- On-demand conversions are low priority. They are meant to allow the microcontroller to perform measurements or diagnostic at chosen time instants. They cannot co-exist with cyclic conversions: to run an on-demand conversion, cyclic conversions must be disabled and the MCU must wait for their termination (monitor the DUTY_ON flag). On the other hand, on-demand conversions cannot interrupt themselves or a configuration override.
- Cyclic conversions are low priority. Their main purpose is to monitor the battery pack and the L9963E status. However, they can also be used to retrieve periodically measurement data. They can be interrupted by the configuration override. They cannot co-exist with on-demand conversions: before enabling cyclic conversions, the MCU must wait for any ongoing on-demand conversion to end first (monitor the DUTY_ON flag).

The following FSM describes the functionality and the transitions among the different operating modes of the voltage conversion routine.

Figure 12. Finite state machine of the voltage conversion routine

SOC = state of conversion



To start on-demand conversions, the user must set SOC = 1 in the ADCV_CONV register: in case the coulomb counting routine is enabled, every time an on-demand voltage conversion is requested by setting SOC = 1, the actual conversion start is delayed until the first useful current conversion takes place. This allows a perfect synchronization between voltage and current samples but might result in a maximum delay of $T_{CYCLEADC\_CUR}$, which must be taken into account by user SW and added to the recommended T_{DATA_READY} as per the device datasheet.

To start cyclic conversions, the user must set CONF_CYCLIC_EN = 1 in the ADCV_CONV register. The ADC_FILTER_CYCLE determines the duration of the routine steps.

Cyclic conversions can be used for diagnostic and measurement purposes:

- In case the routine is only intended for diagnostic purposes, the user can set CYCLIC_UPDATE = 0. This setting causes any conversion result to be used only for internal comparisons. Data are subsequently discarded and registers containing measurement results are not updated.
- In case measurement results are used for diagnostics and SOC estimation, the user must set CYCLIC_UPDATE = 1, thus causing measurement register update upon each step completion, as for on-demand conversions. Note: results of a previous on-demand conversion might be overwritten by the ones of cyclic executions.

Two counters are implemented for driving the cyclic execution:

- TCYCLE, which is an SPI programmable timer accounting for cycle period.
- NCYCLE, which is an internal counter, incremented by 1 every time TCYCLE expires: it counts the number of cycles executed.

These counters are started/stopped upon FSM transitions.

4 Coulomb counting routine

The coulomb counting routine is performed to evaluate the charge injected/subtracted during vehicle operation.

To enable it, the CoulombCounter_en bit must be set to 1.

Disabling the coulomb counter by setting CoulombCounter_en to 0 does not reset the accumulator (CoulombCounter_msb, CoulombCounter_lsb) and sample counter (CoulombCntTime) registers. The MCU is responsible for resetting the coulomb counter, clearing any data previously stored.

5 Cell current measurement

The current flowing into the external shunt resistance RSENSE is measured through a differential amplifier stage (connected between ISENSEP/ISENSEM pins) feeding a 18-bit ADC.

The current conversion chain can be enabled through the **CoulombCounter_en** bit and runs in background to perform the **Coulomb Counting Routine**.

Moreover, L9963E also allows to synchronize the **Voltage Conversion Routine** and the **Coulomb Counting Routine** for a precise State Of Charge estimation. Everytime an on-demand voltage conversion is requested by setting **SOC = 1**, the actual conversion start is delayed until the first useful current conversion takes place. This might result in a maximum delay of TCYCLEADC_CUR, that must be taken into account by user SW only in case current ADC is enabled

6 L9963E safety and diagnostic features

The L9963E provides an extended set of safety mechanisms to reach the required ASIL standard. It monitors potentially damaging conditions for the battery pack.

6.1 Cell UV/OV diagnostic

It is possible to select the value for the overvoltage threshold as well as for the undervoltage threshold of the cells.

This diagnostic feature is completed by analyzing, inside the logic block, the digital information provided by the voltage measurement ADCs. Measurements are performed just on enabled cells.

In case of cell UV/OV (VCELL_OV/UV):

- The corresponding fault flag is set and latched in the VCELL_OV / VCELL_UV register.
- Fault is propagated through the FAULT line.
- Balancing is stopped in case of UV event:
  - A cell UV causes balancing to be stopped on the whole cell stack.
  - A cell balance UV causes balancing activity to be stopped only on the affected cell.
  - The conversion routine goes into configuration override.

6.2 Total battery VBAT diagnostic

The total stack voltage diagnostic is implemented through three different safety mechanisms:

- Arithmetic sum of the digital information of cell ADC (within the cell conversion step of the voltage conversion routine). Such a value is then compared to the digital thresholds (programmable via registers).
- Direct conversion of the voltage at VBAT pin through internal resistive divider. The result is compared to fixed thresholds. This diagnostic is mainly intended to protect the IC against absolute maximum rating (AMR) violation on the VBAT pin. It can also be used as a redundant coherency check with the arithmetic sum of cells.
- Continuous sense of the VBAT pin voltage with a VBAT_UV/OV comparator, featuring fixed thresholds. It is used as an overvoltage warning or an undervoltage warning. This diagnostic is intended to provide a fast reaction against transient overvoltage and undervoltage events.

This UV/OV comparator is always enabled to guarantee a continuous safety check on VBAT voltage.

6.3 VBAT overvoltage

The aim of this diagnostic is to detect a dangerous increase of battery voltage to protect the circuitry connected to VBAT.

If VBAT > VBAT_OV_WARNING (COMP) (for a time longer than TVBAT_FILT) or VBATT_SUM > VBAT_OV (SUM) or VBATT_MONITOR > VBAT_CRITICAL_OV_TH, the overvoltage fault is directly reported in registers and notified to the microcontroller with 3 dedicated flags, according to the Fault communication procedure.

6.4 VBAT undervoltage

The aim of this diagnostic is to detect a decrease of battery voltage to notify this fault that may cause system malfunctions.

6.5 Cell open wire diagnostic

The open cell detection can be performed through the voltage conversion routine. The diagnostic strategy depends on the ADC_CROSS_CHECK bit.

6.5.1 Cell open with ADC_CROSS_CHECK = 0

If the cell terminal diagnostics step of the voltage conversion routine is executed having programmed ADC_CROSS_CHECK = 0, then the diagnostic addresses the following failures:

- RLPF (low pass filter resistor) degradation: diagnostic has been implemented to guarantee that low pass filter resistor in series to the Cx pin is below the critical limit RLPF_OPEN:
  - On odd cells, RLPF degradation will cause the assertion of the corresponding CELLx_OPEN flag.
  - On even cells, flag assertion depends on the RLPF degradation:
    - Small degradation (RLPF < 24 kΩ typ. with 10 nF CLPF) only causes the assertion of the corresponding CELLx_OPEN flag.
    - Large degradation (RLPF > 24 kΩ typ. with 10 nF CLPF) causes the assertion of both the corresponding CELLx_OPEN flag and the lower odd cell CELLx-1_OPEN flag.
- L9963E C1-C14 pin open.
- L9963E C0 pin open or PCB connector open.

Diagnostic is present on enabled cells only (VCELLx_EN = 1).

6.5.2 Cell open with ADC_CROSS_CHECK = 1

ADC_CROSS_CHECK = 1, then the diagnostic addresses the following issues:

- Failure in the filtering capacitor CLPF (low pass filter capacitor) causing an excessive leakage from cell.
- ADC error due to bandgap shift or failure on the conversion path.

For each pair of consecutive cells, the two corresponding ADCs, each of whom is referenced to a different bandgap, are measuring the voltage drop on the external RLPF.

Since no pull-down current is applied while measurement is ongoing, the voltage drop on RLPF is expected to be null, and the two measurement results should match.

If one of the two ADCs is experiencing an issue, or an excessive leakage from the CLPF is causing a voltage drop on the RLPF, a mismatch in the results occurs. If such a mismatch is greater than VADC_CROSS_FAIL, failure is detected.

6.6 PCB open diagnostic

To detect loss of cell wire at PCB connector, follow the procedure described in the L9963E datasheet.

Note: when performing PCB open diagnostic, other diagnostics such as Cell UV/OV diagnostic and balancing open load diagnostic might also be triggered. They must be then discarded by user software.

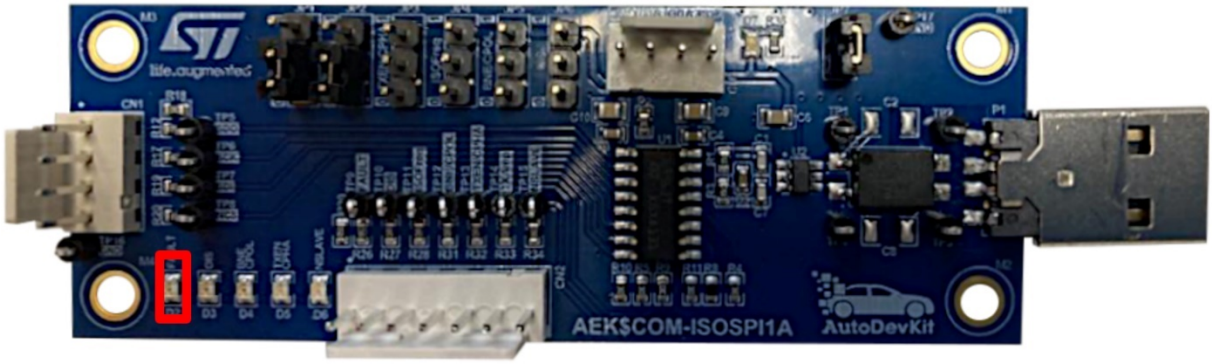
6.7 Die temperature diagnostic and overtemperature

An internal temperature sensor continuously monitors the temperature of the chip.

The chip prevents overheating through an overtemperature threshold TSD (which includes a hysteresis TSD_HY). Once the die temperature reaches TSD, a thermal shutdown circuit will force the chip to reduce the consumption by stopping balancing. A fault is reported to the microcontroller with a dedicated bit OTchip and propagated through the FAULT Line. When the temperature of the die returns to a normal level, L9963E can resume the normal operation. Balancing is released after the microcontroller reads OTchip latch.

7 Fault condition in daisy chains

Figure 13. **Fault LED on the AEK-COM-ISOSPI1**



When using the AEK-POW-BMS63EN with the AEK-COM-ISOSPI1 transceiver board, you can detect a fault through the fault LED of the image above. It is related to the state of all the BMS nodes in the daisy chain. If an undervoltage, overvoltage, overcurrent, or overtemperature occurs on any cell of a BMS, a fault condition is detected. To solve this condition, diagnosis via software code must be activated.

The overcurrent detection is linked to a threshold defined in the application, not in the software driver. The threshold must be modified according to the load.

8 L9963E cell balancing

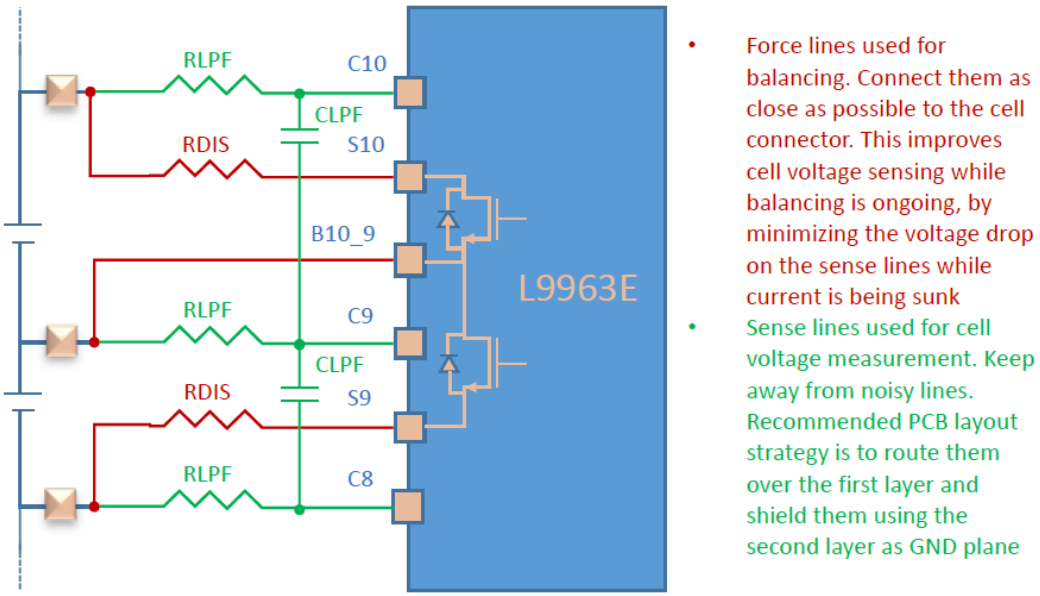
In the L9963E, the Sx and Bx_x-1 pins are used to balance the charge of the cells by discharging the ones with a higher SOC. Balancing can be performed either with external resistors or internal MOSFETs.

Cell balance drivers are powered by VBAT stack voltage. Hence, balancing is theoretically possible even at low cell voltages, except for cell 14. In case $V_{CELL14} < V_{CELL14\_BAL\_MIN}$, the corresponding balancing circuitry does not operate properly, and false overcurrent detection may occur.

8.1 Passive cell balancing with internal MOSFETs

The board is designed using internal MOSFETs.

Figure 14. Cell monitoring with internal balancing



The on-chip MOSFETs are switched on to sink a current from the cell, thus dissipating charge on RDIS. The affordable balancing current is restricted by the thermal relief on the current source circuits.

The maximum balance current on each cell is 200 mA. All cells can be balanced simultaneously, if the junction temperature does not exceed the maximum operating defined in the datasheet. To prevent thermal overstress, the die temperature diagnostic and overtemperature protections are implemented.

9 Methods for SOC estimation

The state of charge (SOC) of a cell indicates the energy stored by a cell, defined as the ratio of the charge quantity of a cell in a given moment and its total capacity.

The traditional calculation method is the coulomb counting.

According to this method, a charging/discharging current I(t) is integrated in time to calculate the residual energy quantity of the cell itself.

SOC(t) in a specific instant is calculated as the sum of the SOC related to the previous instant SOC(t-1) and the quantity of energy removed/added to that specific instant, as shown in the following formula:

$$SOC_{(t)} = SOC_{(t-1)} \cdot \int_{t-1}^{t} \frac{I_t \cdot \eta}{Q_n} \cdot dt$$

where:

- SOC(t) is the state of charge of the cell in the instant (t)
- SOC(t-1) is the state of charge of the cell in the previous instant (t-1)
- Q is the total capacity of the cell
- η is the cell efficiency

This method does not consider the speed, and hence the dynamics of the cell charge/discharge, temperature, hysteresis of charge/discharge and especially the cell age, which directly influences efficiency.

Moreover, this method is not accurate due to the error of integration and to the initial SOC estimation, and to the eventual inaccuracy of the cell current measurement.

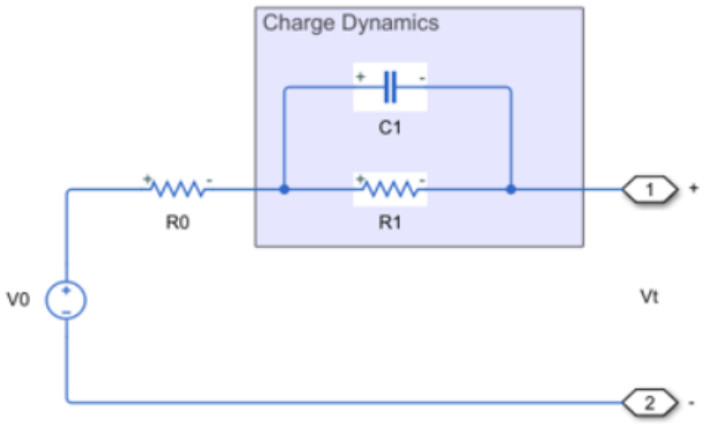
Other issues are related to the difficult estimation in case the cell undergoes an open circuit voltage variation (OCV with no load voltage) that is not significant in relation to the actual SOC (for example, in case LFP batteries are used).

The Kalman filter represents a promising alternative approach, which allows overcoming these obstacles through a higher effort in terms of computational resources.

This filter is based on a non-linear model of a cell through an equivalent circuit.

In our approach, we have used a first-grade equivalent circuit, as shown in the image below.

Figure 15. Equivalent circuit



In this circuit:

- Vo is the no-load voltage
- Vt is the voltage at the battery poles
- R0 represents the ohmic behavior of the cell
- R1C1 takes into account the dynamic behavior during charge and discharge

The equations that define the model are:

$$\frac{dSOC}{dt} = -\frac{i}{3600AH(T)}$$

$$\frac{dV_1}{dt} = \frac{i}{C_1(SOC,T)} - \frac{V_1}{R_1(SOC,T)\cdot C_1(SOC,T)}$$

$$V_t = V_0(SOC,T) - iR_0 - V_1$$

Where:

- SOC is the state of charge
- i is the current
- V_0 is the no-load voltage
- V_t is the terminal voltage
- AH is the ampere-hour rating
- R_1 is the first polarization resistance
- C_1 is the parallel RC capacitance
- T is the temperature

As opposed to the coulomb counting approach, this model considers the voltage at the battery poles and the cell capacity, as well as the temperatures and charge/discharge dynamics.

The equivalent circuit parameters used in the equations above must be estimated during the characterization phase.

Building the Kalman filter requires the definition of the state to estimate and the functions of process and observation, as shown below.

$$x = [SOCV_1]^T$$

$$f_{(x,i)} = \begin{bmatrix} -\frac{i}{3600AH(t)} \\ -\frac{i}{C_1(SOC,T)} - \frac{V_1}{R_1(SOC,T)\cdot C_1(SOC,T)} \end{bmatrix}$$

$$h(x,i) = V_0(SOC,T) - iR_0 - V_1$$

The Kalman filter has been configured in Simulink according to the above functions.

The parameters (Vo, R0, C1, and R1) have been configured through lookup tables, after characterizing an INR 18650 MJ1 battery by LG.

The x status matrix (see above) estimated by the Kalman filter can define the SOC over time during the charge/discharge operations and at no load.

From the Kalman model in Simulink, we generated the C code through the embedded coder tool. Then, the code has been integrated into the L9963E drivers to estimate the SOC in real-time. For further details, see the related Mathworks page.

Another method to estimate the SOC is based on neural networks. This method requires a network training phase in all the operative conditions of the cell that belongs to the battery pack (during the charge/discharge operations and at no load).

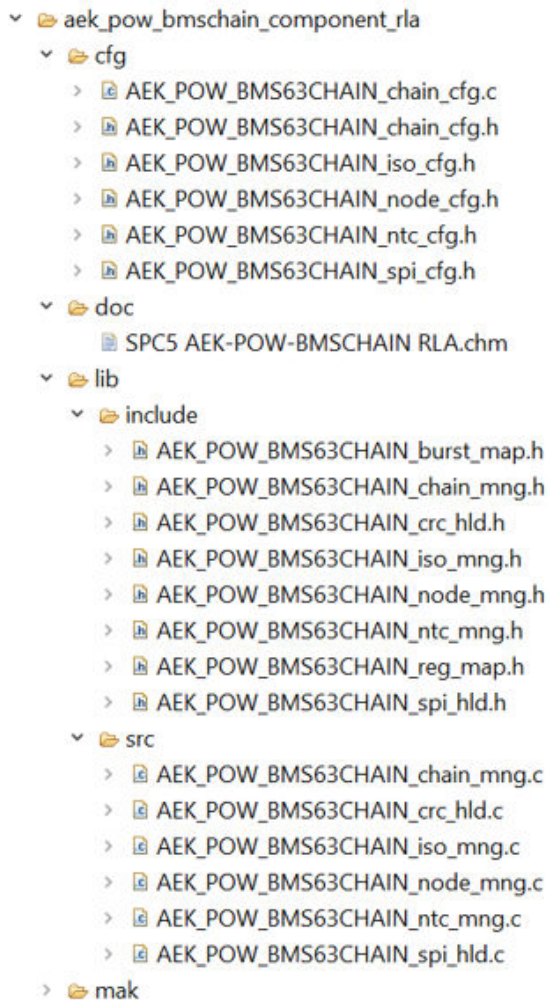
10 AutoDevKit ecosystem

The application development employing our BMS evaluation boards takes full advantage of the AutoDevKit ecosystem, whose basic components are:

- AutoDevKit Studio IDE (STSW-AUTODEVKIT)
- OpenOCD programmer and debugger

10.1 AEK-POW-BMSCHAIN component folder structure

Figure 16. AEK-POW-BMSCHAIN component folder structure



The cfg folder contains all the configuration files.

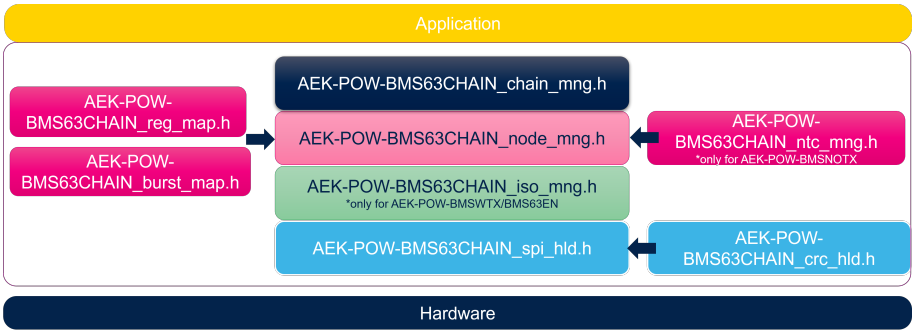
The doc folder contains the doxygen documentation.

The lib folder contains the component header and source files.

10.2 Software component architecture (AEK-POW-BMSCHAIN component RLA)

The following image shows the architecture of the software components created for our BMS boards, which consists of the layers shown in the image below:

Figure 17. **Software architecture**



SPI High Level Driver contains APIs that sends SPI message packets to the daisy chain. These functions implement SPI communication in Single Access and Broadcast Access to read and write L9963E registers in each node.

CRC High Level Driver contains APIs that implement CRC calculation for packets sent via SPI.

ISOSPI manager contains APIs that handle the AEK-POW-ISOSPI1 transceiver in a daisy chain in Single or Dual Access Ring configurations. If the first chain node is an AEK-POW-BMSNOTX evaluation board, the ISOSPI manager is not used.

Node Manager contains APIs that use SPI High Level Driver to implement measure and diagnostic functions for the chain BMS nodes through L9963E register reading and writing.

reg_map-h contains the map of L9963 registers for each chain node.

Burst_map contains the map of burst commands and replies from L9963 after SPI requests.

NTC Manager contains APIs to handle external NTCs for the AEK-POW-BMSNOTX evaluation board. This software manager is excluded from compiling in case there is no AEK-POW-BMSNOTX evaluation board in daisy chains.

Chain manager represents the driver entry level and implements APIs that handle one or more allocated BMS chains. These APIs include the function for addressing procedure and parameter setting defined in the component.

10.3 How to import the AEK-POW-BMSCHAIN component in AutoDevKit

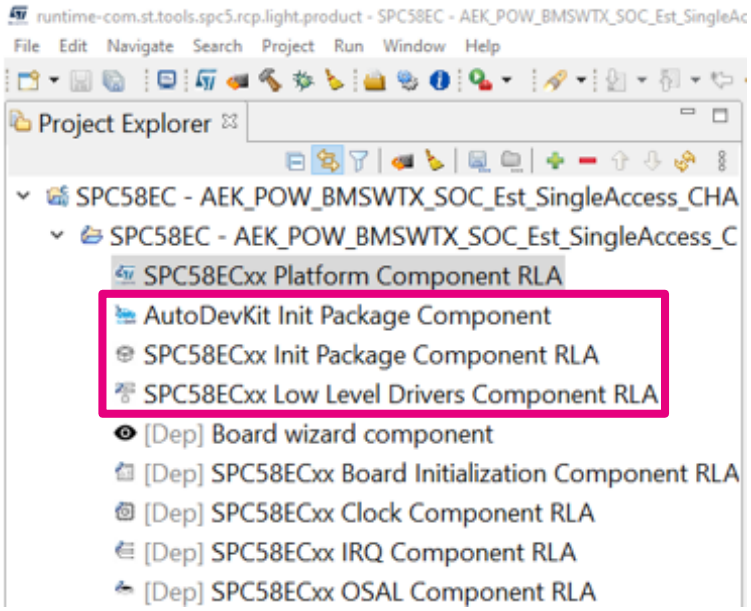
This example is an application for a centralized daisy chain topology using the AEK-MCU-C4MLIT1 as the microcontroller board. To recreate this scenario, follow the procedure below.

Step 1. Create a new SPC5-STUDIO application for the SPC58EC series microcontroller and add the following components:

– SPC58ECxx Init Package Component RLA
– SPC58ECxx Low Level Drivers Component RLA
– AutoDevKit Init Package Component

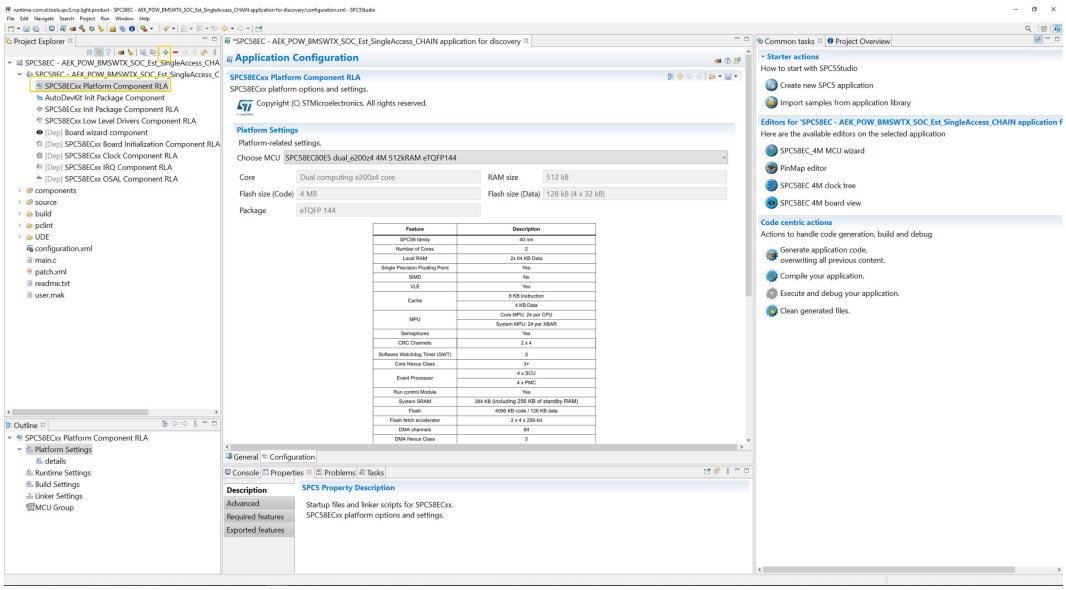
These components must be added immediately, or the other components will not be visible.

Figure 18. Adding components



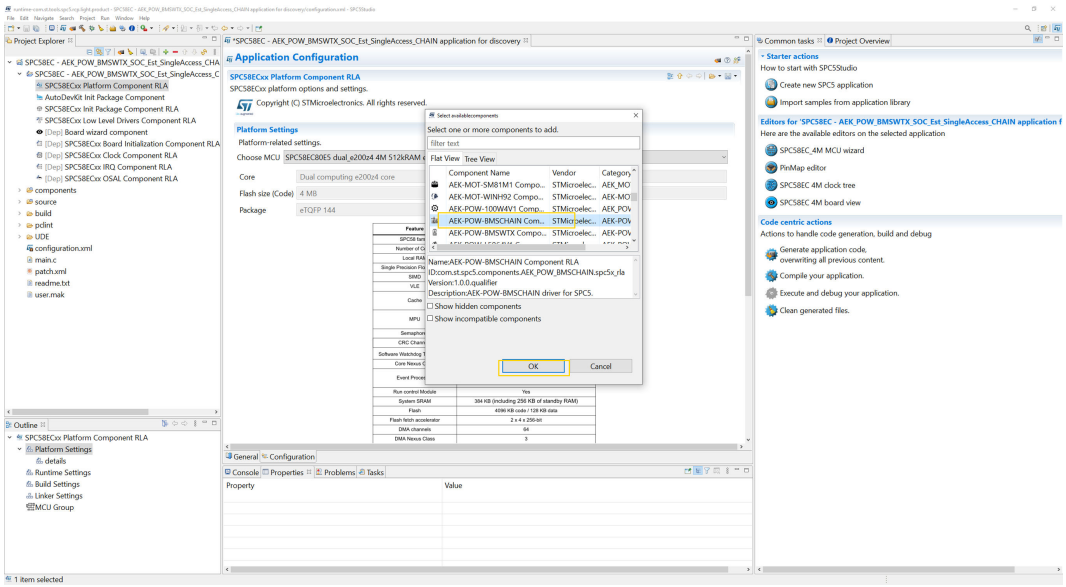
Step 2. Click on SPC58ECxx Platform Component RLA. Then, click on “+” to import the new component.

Figure 19. Adding SPC58ECxx Platform Component RLA



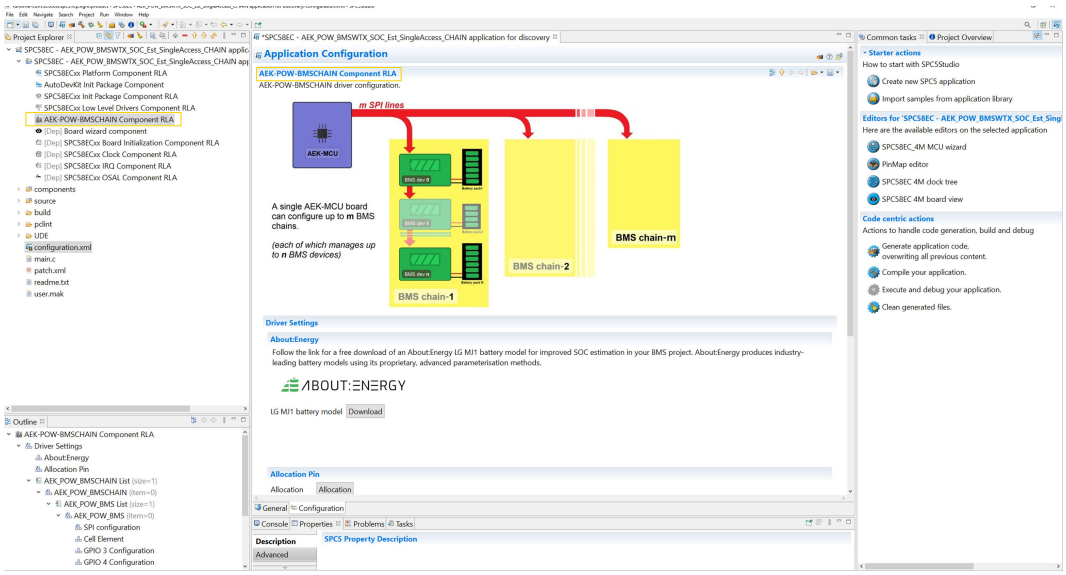
Step 3. Select “AEK-POW-BMSCHAIN”. Then, click on “OK”.

Figure 20. Selecting AEK-POW-BMSCHAIN Component



Step 4. If you have correctly imported the component, it is then visible in the Project explorer section, as highlighted below.

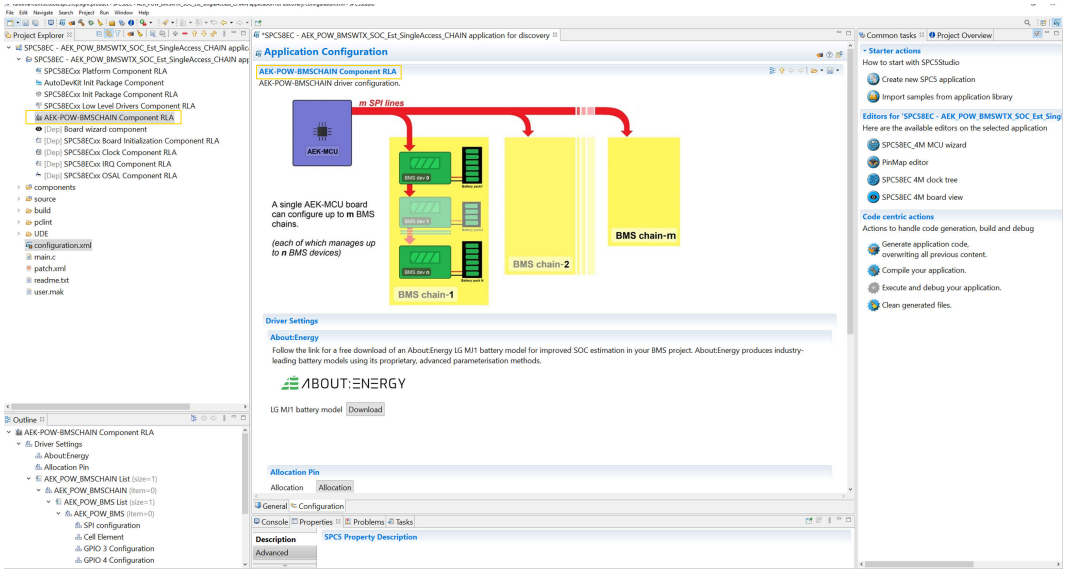
Figure 21. AEK-POW-BMSCHAIN component visualization in the Project explorer and application configuration windows



10.4 Chain allocation

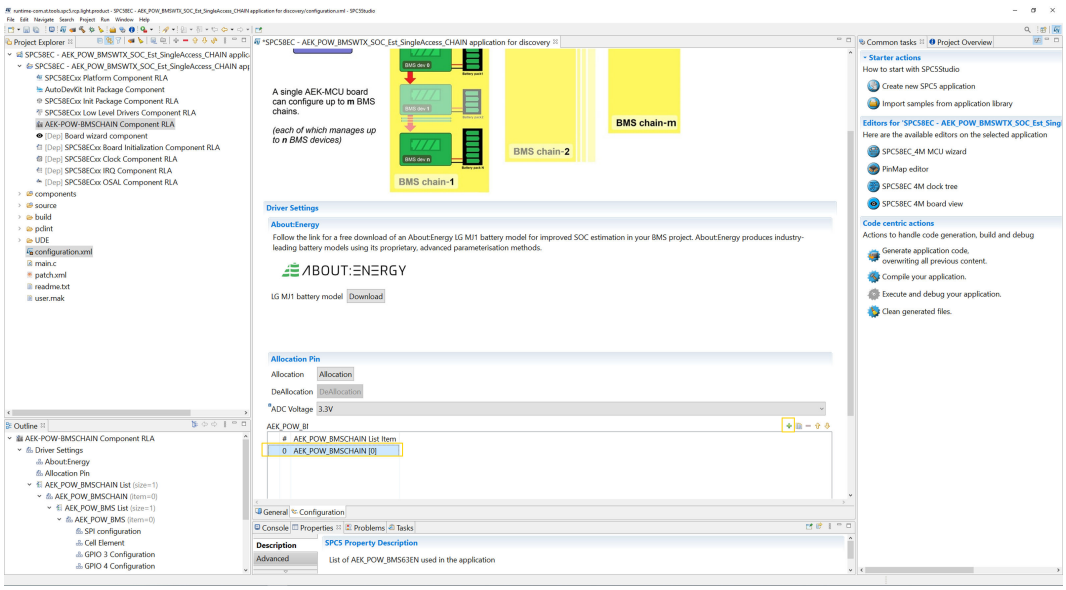
Step 1. Select the AEK-POW-BMSCHAIN component in the Project Explorer section.

Figure 22. AEK-POW-BMSCHAIN component selection



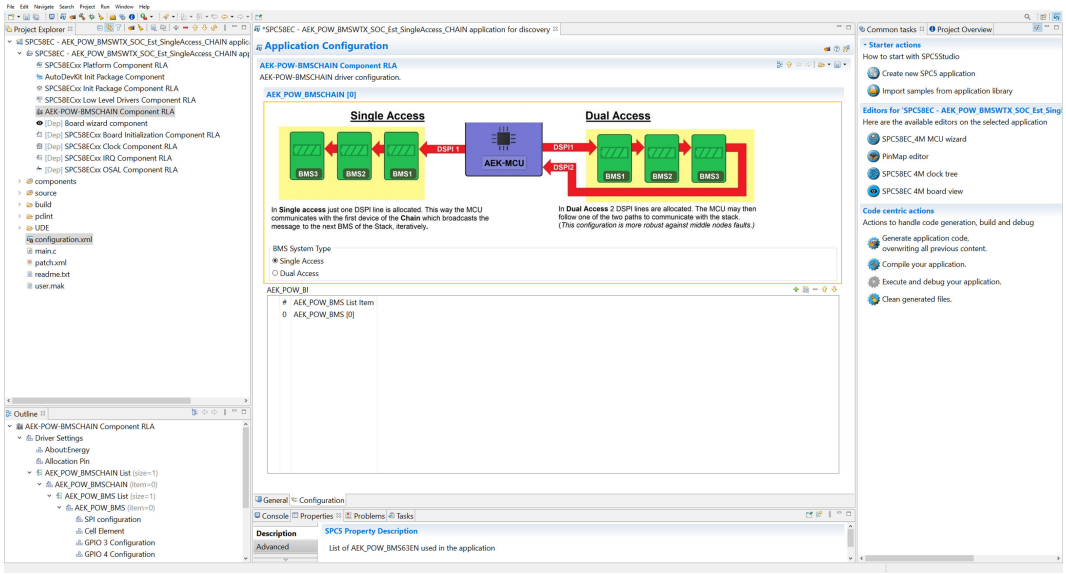
Step 2. Click on “+” to add a new chain to the allocation table. Then, click on the chain to customize it.

Figure 23. Chain customization



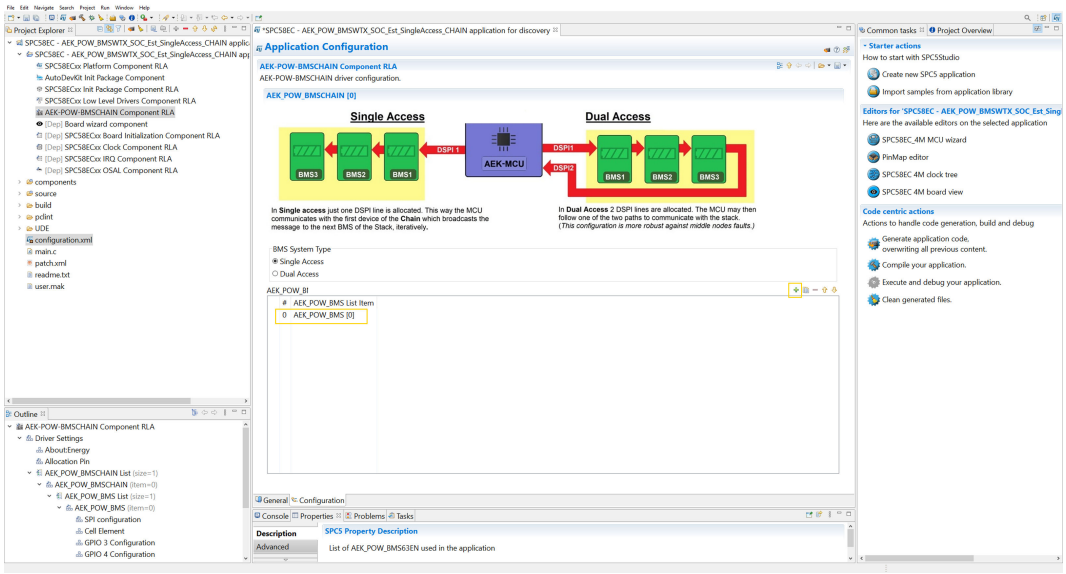
Step 3. Choose a chain topology: “Single Access” (Centralized) or “Dual Access”.

Figure 24. Chain topology selection



Step 4. Click on “+” to add a new node to your chain. Then, click on the newly added node to customize it.

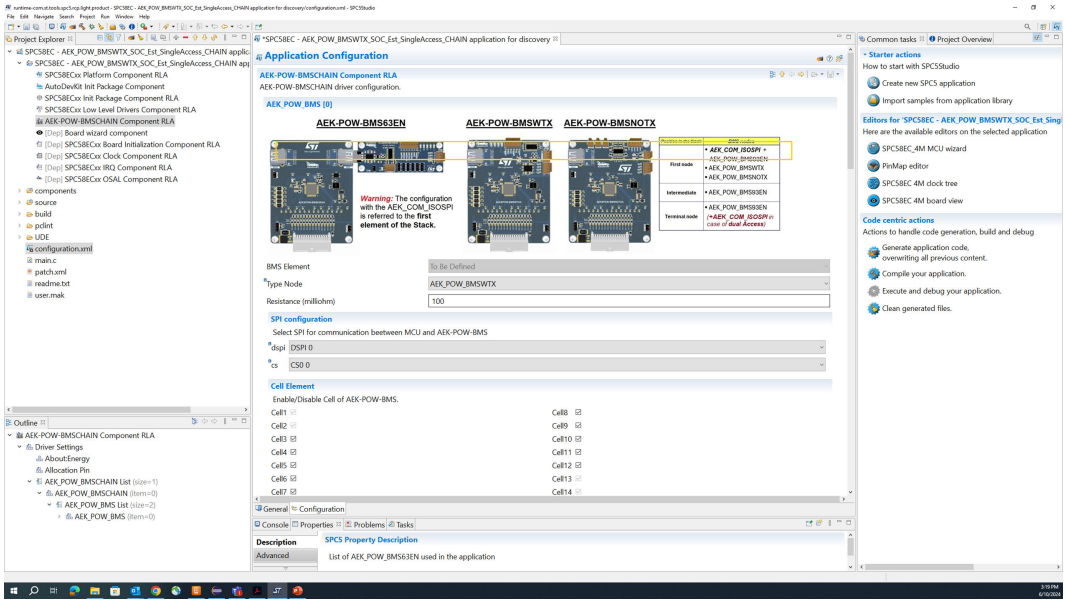
Figure 25. New node addition and customization



Step 5. Each node has its own dedicated configuration page, featuring several options that must be customized or left to their default value. The “BMS Element” field defines the topology of each chain node (First Element, Node or End Element).

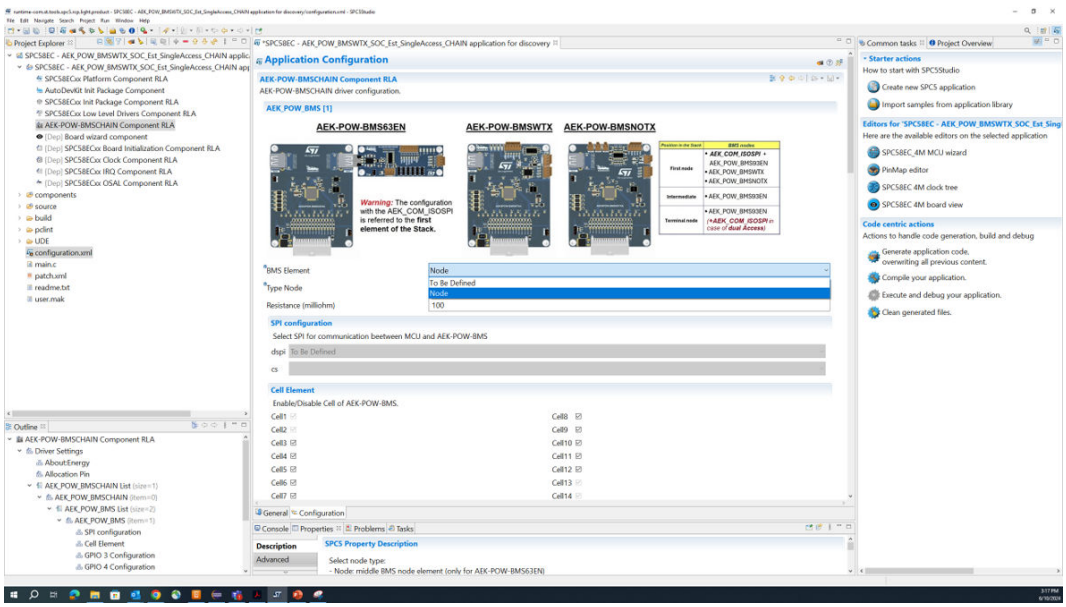
Note: In both Single Access and Dual Access modes, the first chain node is defined as “First Element”. So, the “BMS Element” field is defined by default.

Figure 26. BMS element field customization details



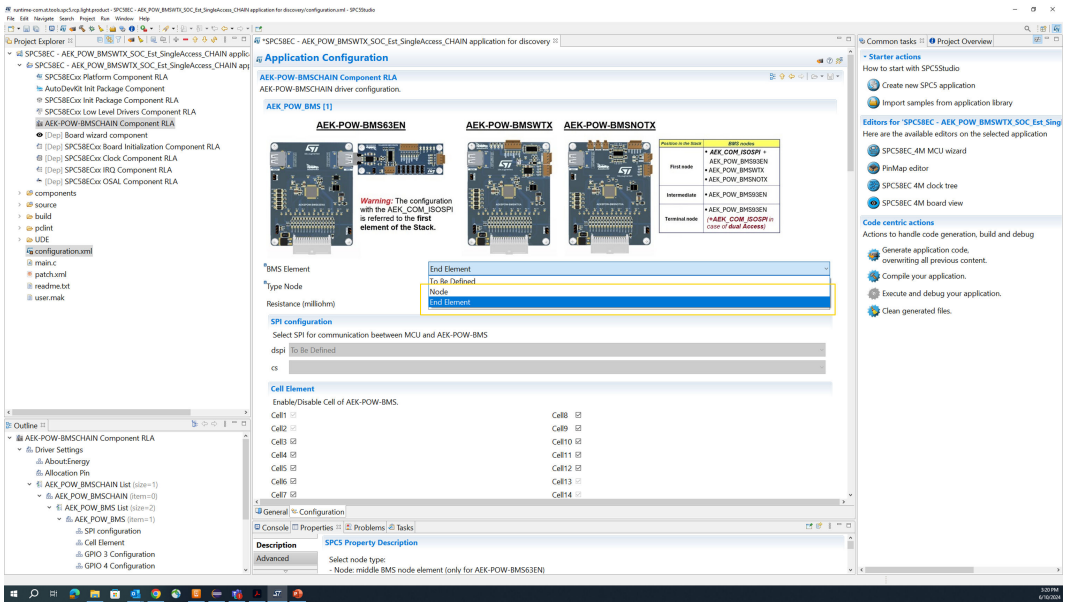
Step 6. If you choose “Single Access” mode chain, fill the “BMS Element” field with “Node” for each node of the chain, except the first (defined as “First Element) and the last (defined as “End Element”) ones.

Figure 27. Single Access mode customization



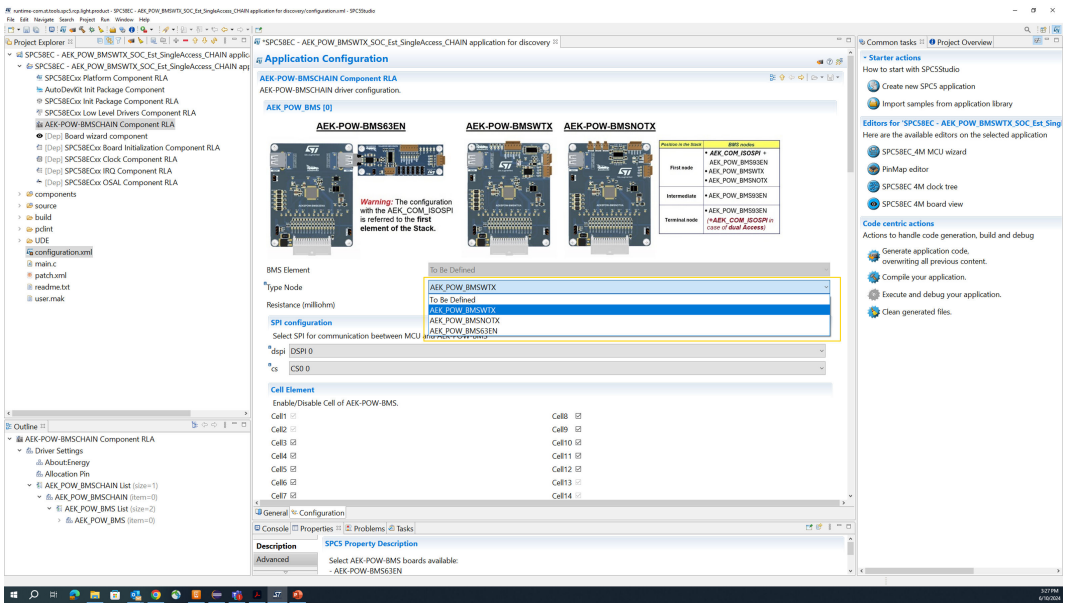
Step 7. If you choose “Dual Access” mode chain, fill the “BMS Element” according to the same rule defined in step 5.

Figure 28. Dual access mode customization



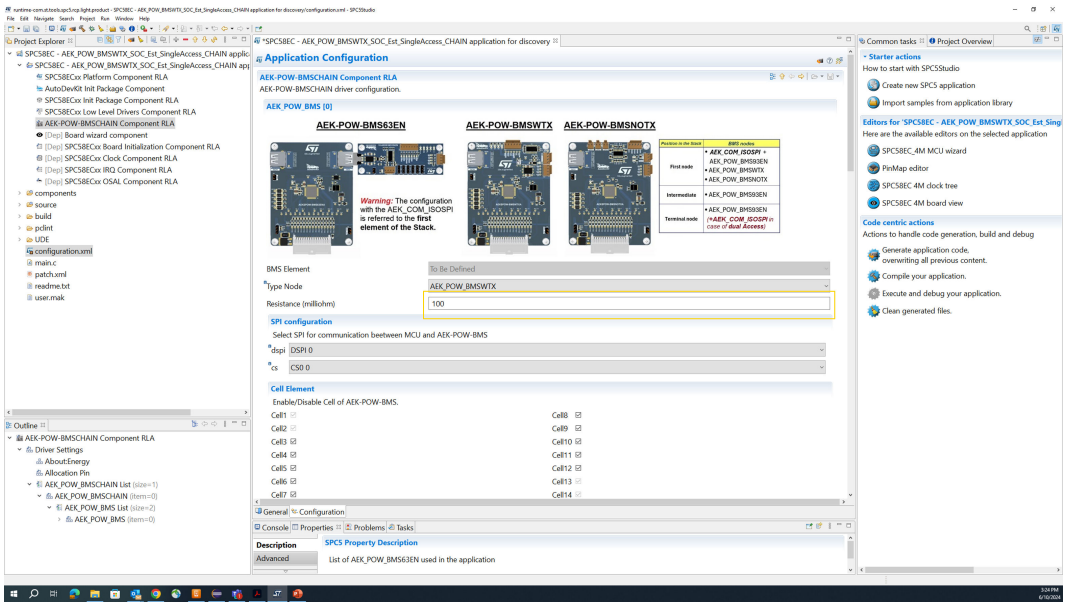
Step 8. Use one among AEK-POW-BMS63EN, AEK-POW-BMSWTX and AEK-POW-NOTX only for the first chain node, both for single access and dual access modes. All remaining nodes must be AEK-POW-BMS63EN to best exploit ISOSPI communication.

Figure 29. Example of chain configuration



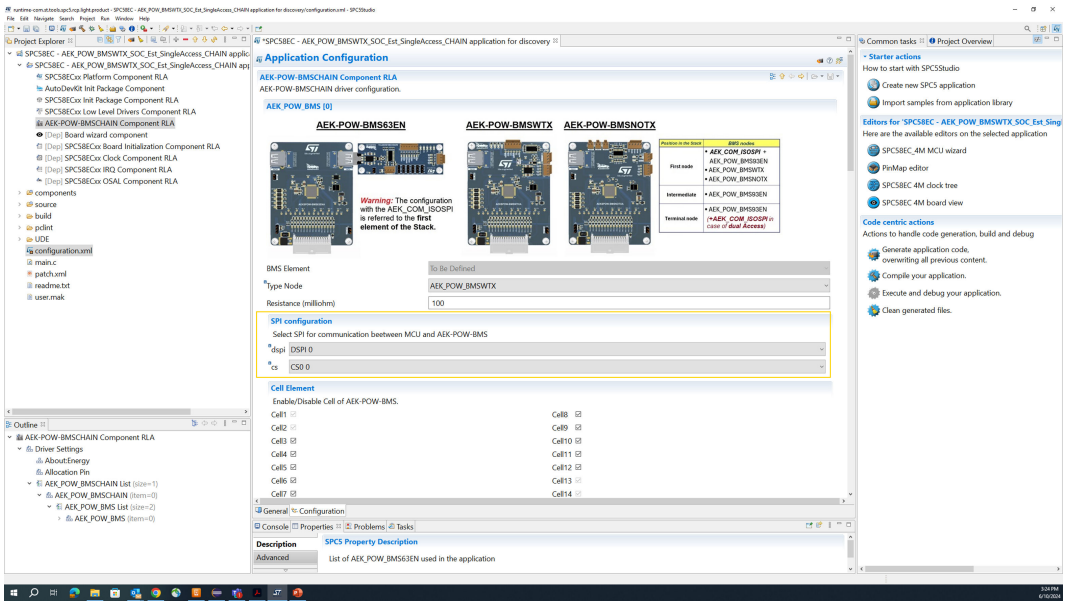
Step 9. Customize the sensing resistor for every single node. The default value is 100 mΩ.

Figure 30. Customizing the sensing resistor



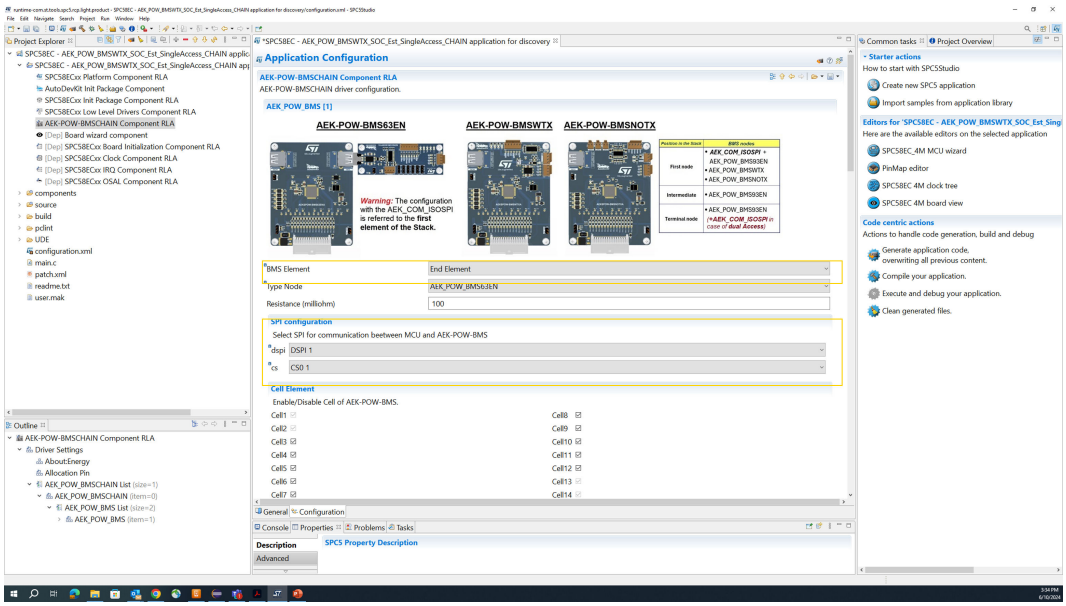
Step 10. In “Single Access” mode chain, define the SPI configuration for the first chain node. Choose the DSPI peripheral and Chip Select.

Figure 31. SPI configuration for single access mode



Step 11. In “Dual Access” mode chain, define the SPI configuration also for the last chain node (“End Element”). Choose the DSPI peripheral and Chip Select for this end node but not for the intermediate nodes.

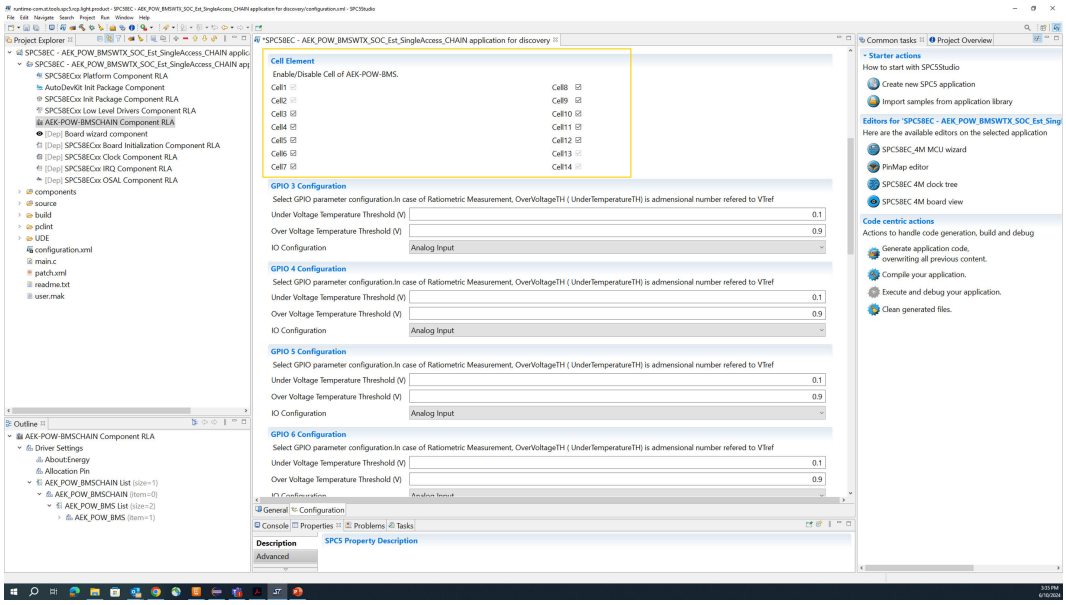
Figure 32. SPI configuration for dual access mode



Step 12. For each chain node, choose and configure the cells to be enabled.

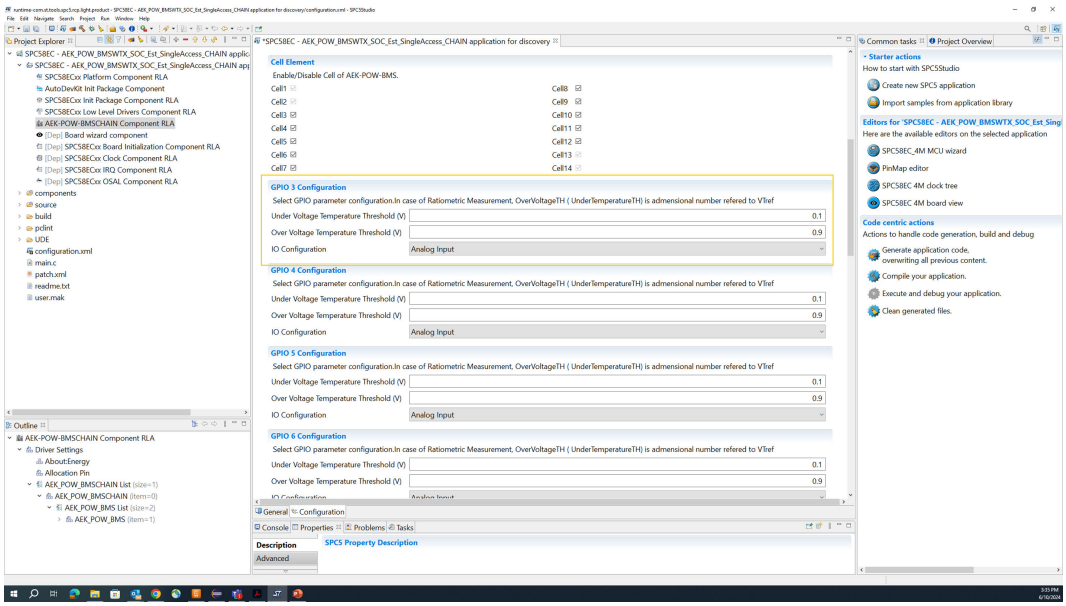
Note: Cell1, Cell2, Cell13, and Cell14 cannot be disabled.

Figure 33. Cell selection and configuration



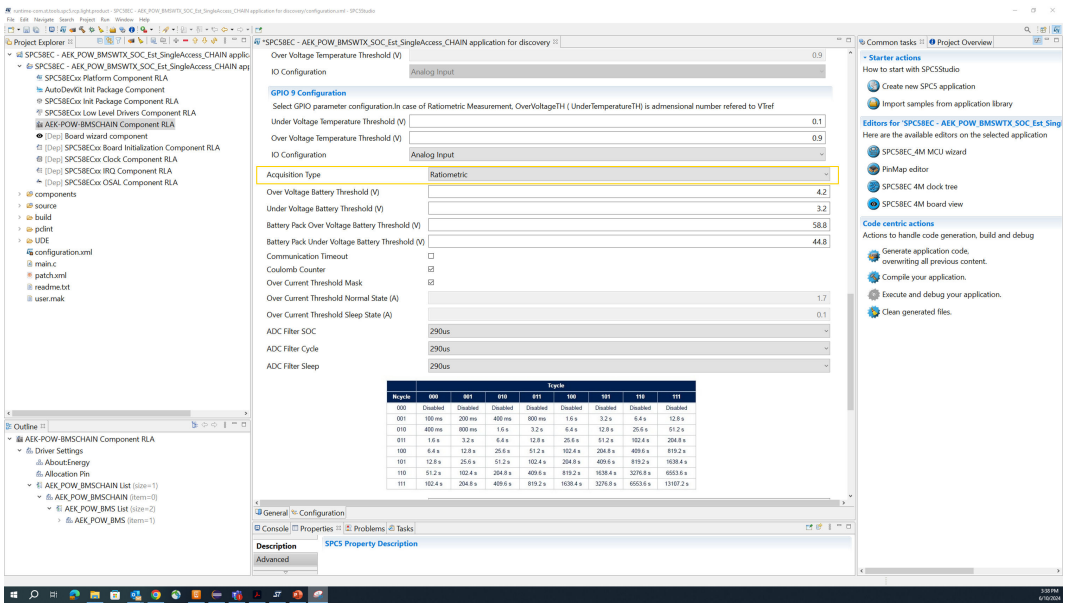
Step 13. Each GPIO handled by the L9963E can be configured as Digital/Analog Input or Digital Output. You can associate temperature thresholds (in voltage) for fault management.

Figure 34. GPIO configuration



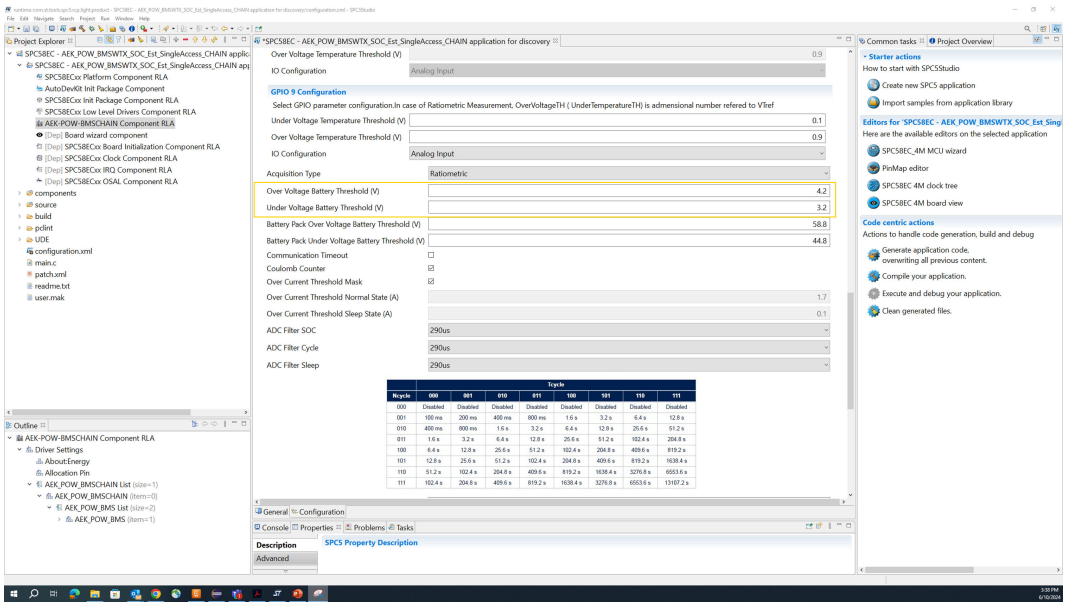
Step 14. For each chain node, configure the GPIO voltage acquisition mode. In Ratiometric mode, voltage is measured according to the L9963E Vtref function. In Absolute mode, measured voltage is not acquired with reference to the Vtref generated by its regulator.

Figure 35. GPIO voltage acquisition mode configuration



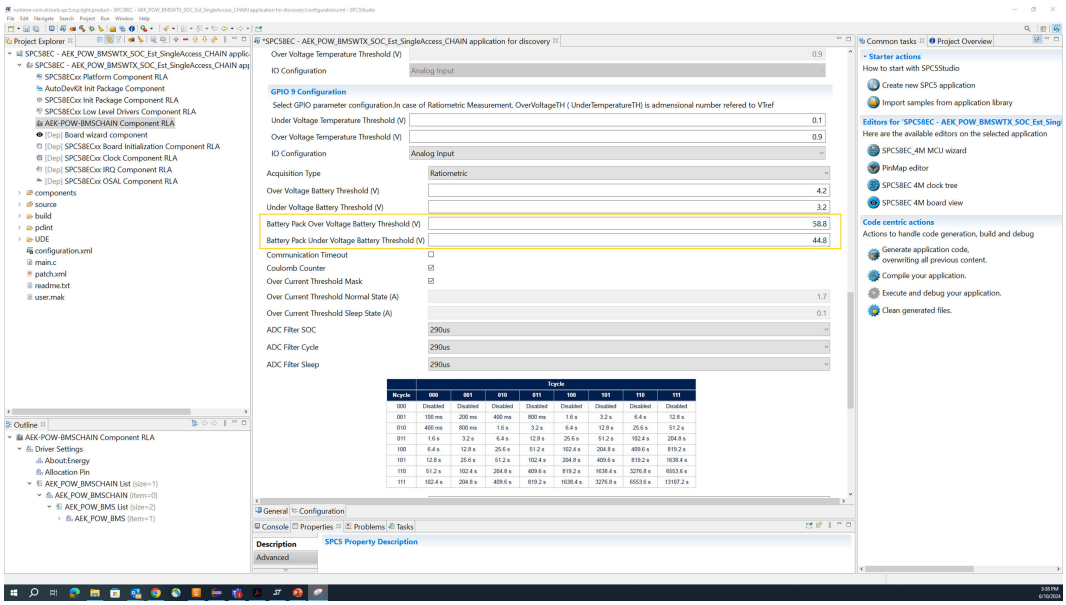
Step 15. Associate voltage thresholds (Under/Over voltage) to each enabled cell. When exceeding the threshold, a fault will be visible if it is not masked via software.

Figure 36. Voltage thresholds association for enabled cells



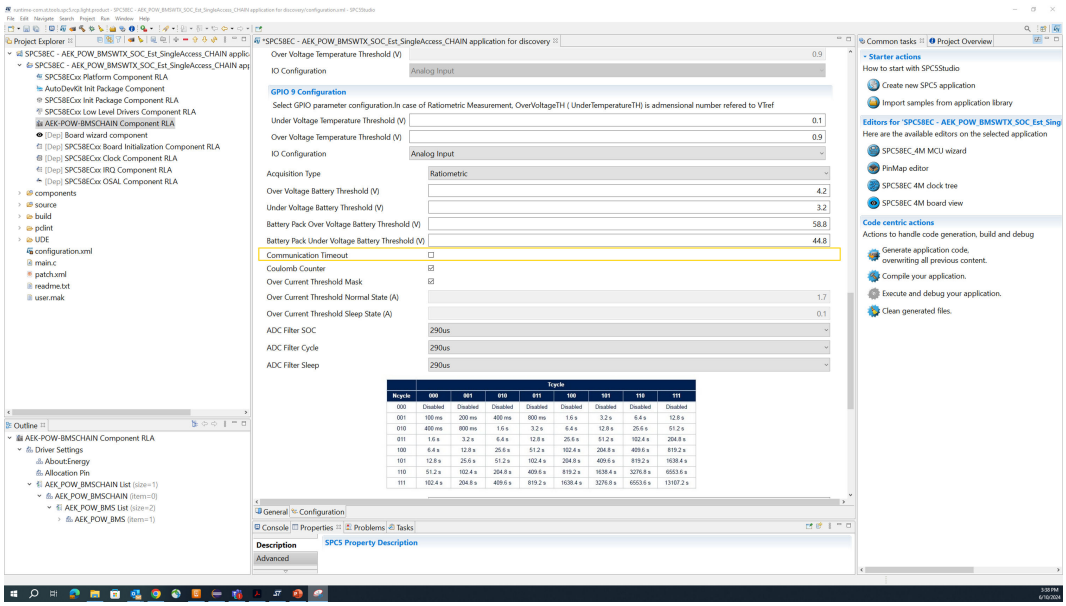
Step 16. You can also associate voltage thresholds (Under/Over voltage) to the entire battery pack managed by each single node. When exceeding the threshold, a fault occurs if it is no masked via software.

Figure 37. Voltage thresholds association for the entire battery pack



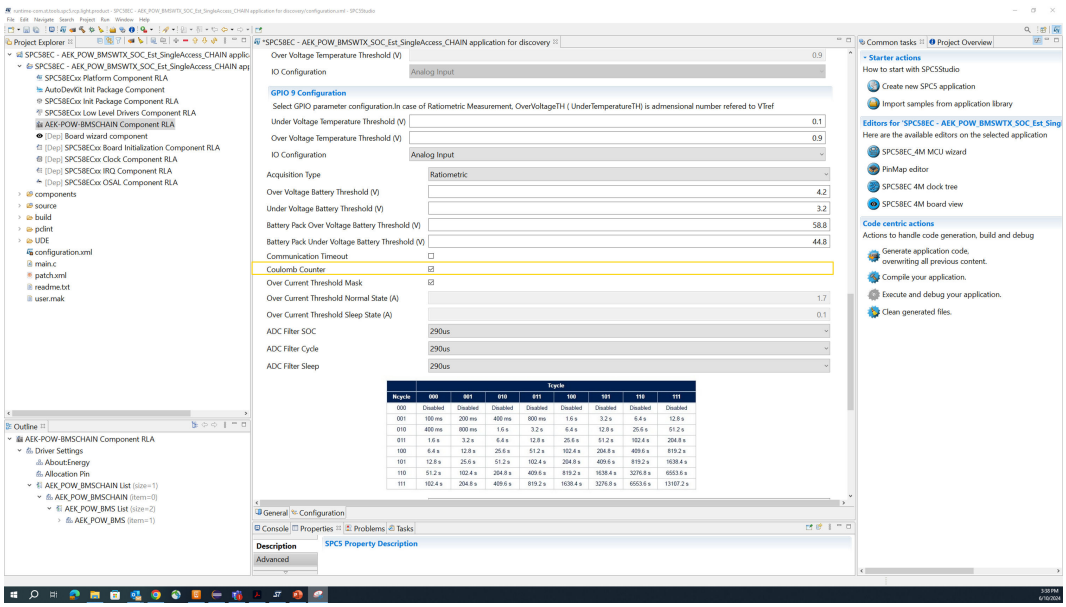
Step 17. When the “Communication timeout” field is enabled, if a chain node does not reply to an SPI command within the time defined in “Communication timeout” (configurable via SPI command), the node enters sleep mode. If the flag is disabled, the node remains in normal mode.

Figure 38. Communication timeout field enablement



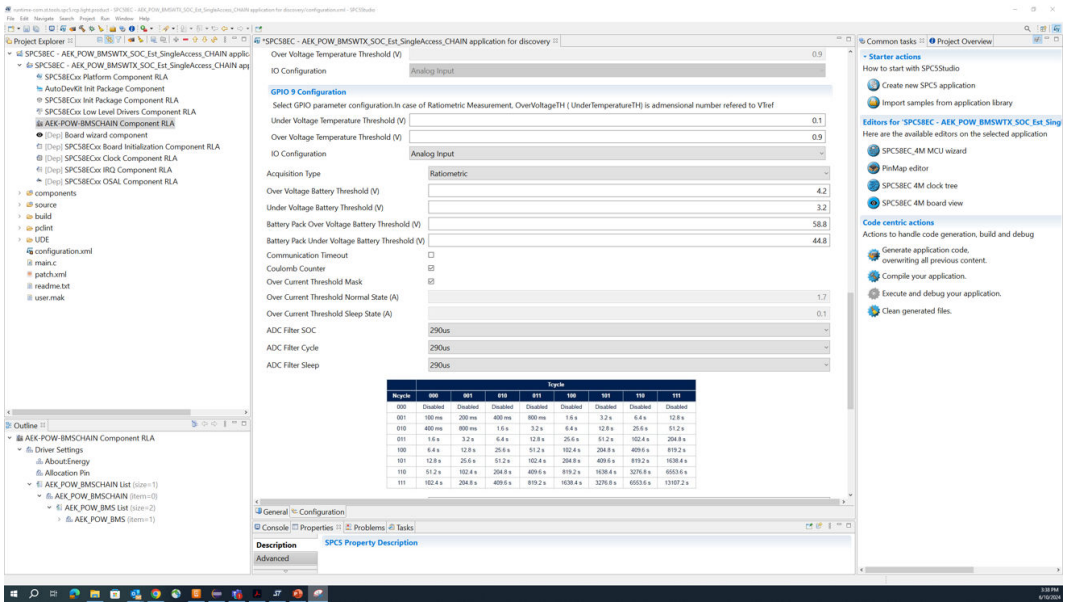
Step 18. If the “Coulomb Counter” field is enabled, values measured from enabled cells will be synchronized with the selected node voltage value.

Figure 39. Coulomb Counter field enablement



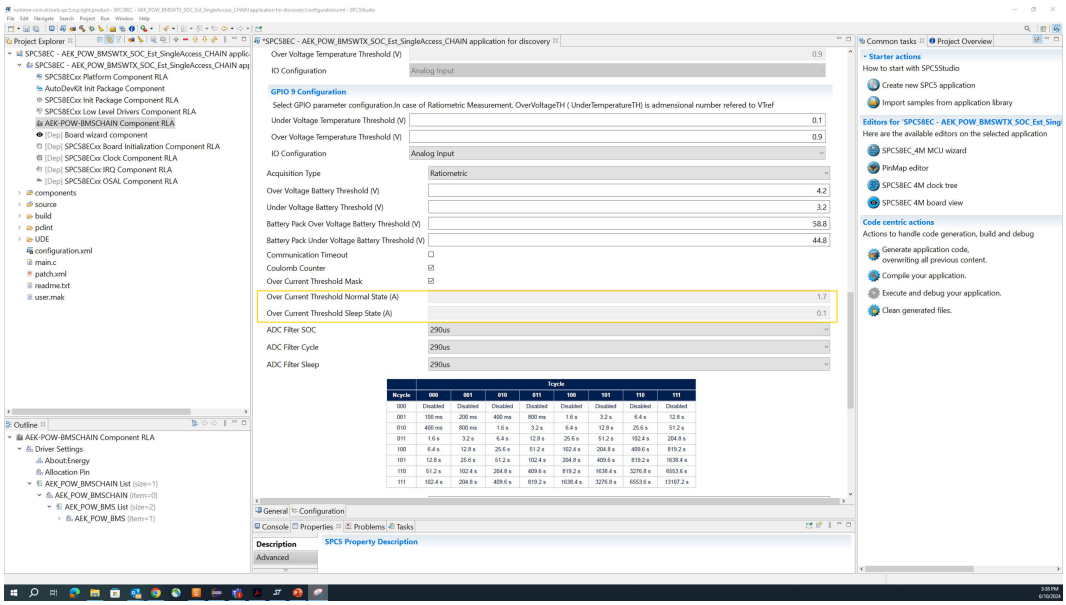
Step 19. If the “Over Current Threshold Mask” field is enabled, the driver masks faults that occur when exceeding the voltage threshold on Rsense, both in sleep and in normal modes.

Figure 40. Over Current Threshold Mask field enablement



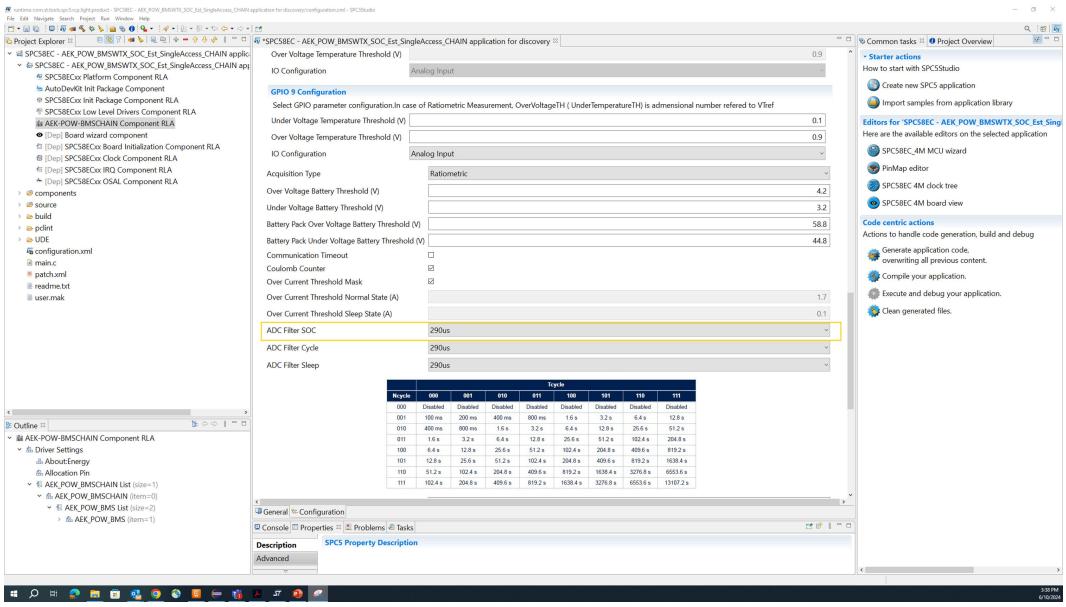
Step 20. Configure overcurrent thresholds in normal and sleep states.

Figure 41. Overcurrent threshold configuration



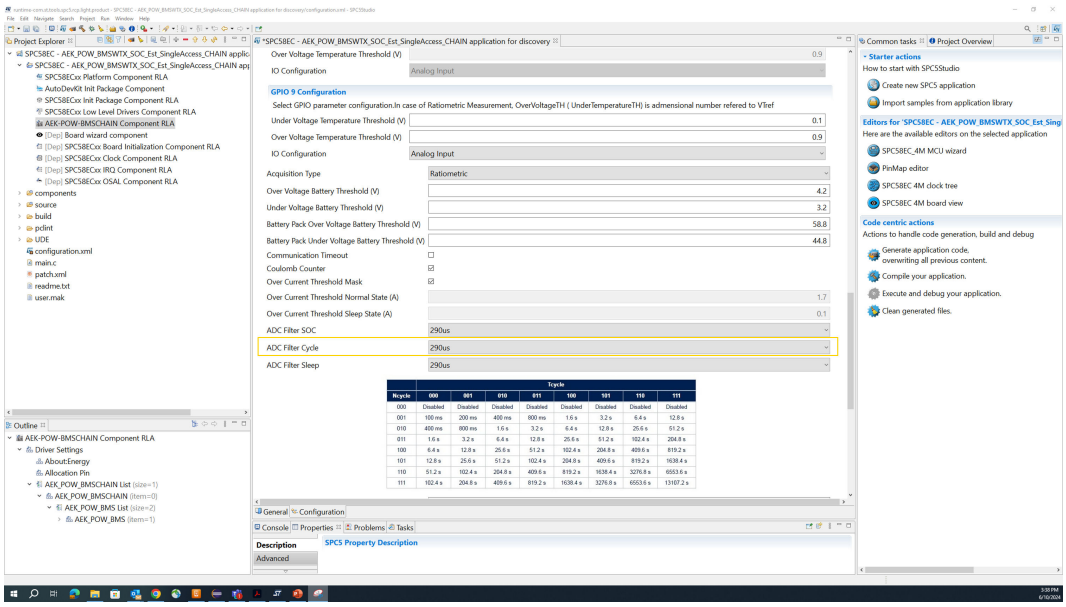
Step 21. Configure “ADC Filter SOC” field to enable on-demand conversion routine (manual request).

Figure 42. ADC Filter SOC field configuration



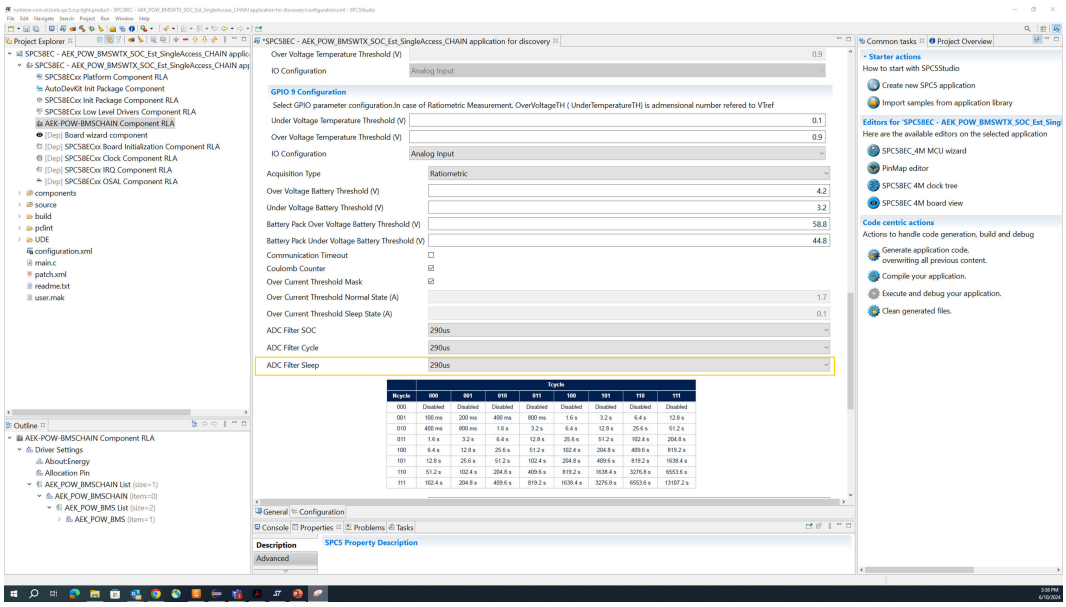
Step 22. Configure “ADC Filter Cycle” field to enable cycle conversion routine in normal state.

Figure 43. ADC Filter Cycle field configuration



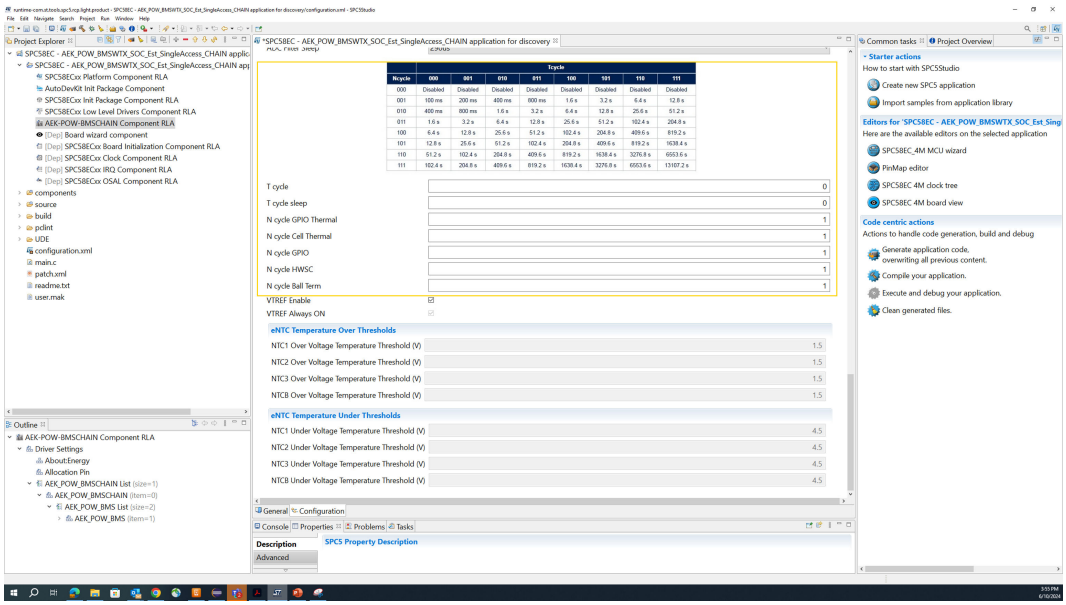
Step 23. Configure “ADC Filter Sleep” field to enable cycle conversion routine in sleep state.

Figure 44. ADC Filter Sleep field configuration



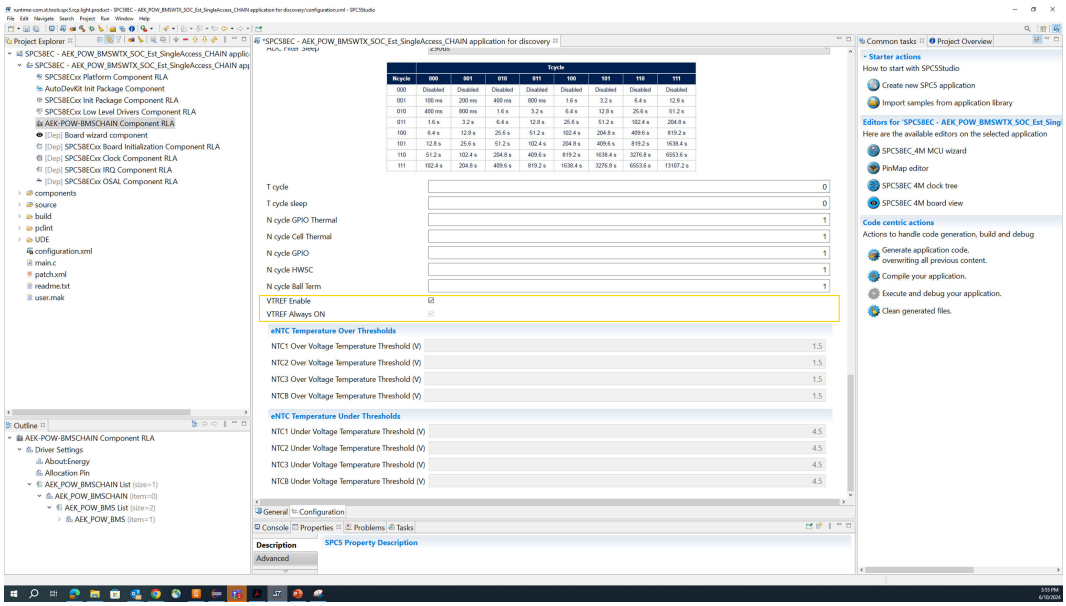
Step 24. Configure periodic acquisition. For example, if Tcycle = 001 and NcycleGPIO = 001, voltage values on GPIOs will be acquired every 200 ms.

Figure 45. Periodic acquisition configuration



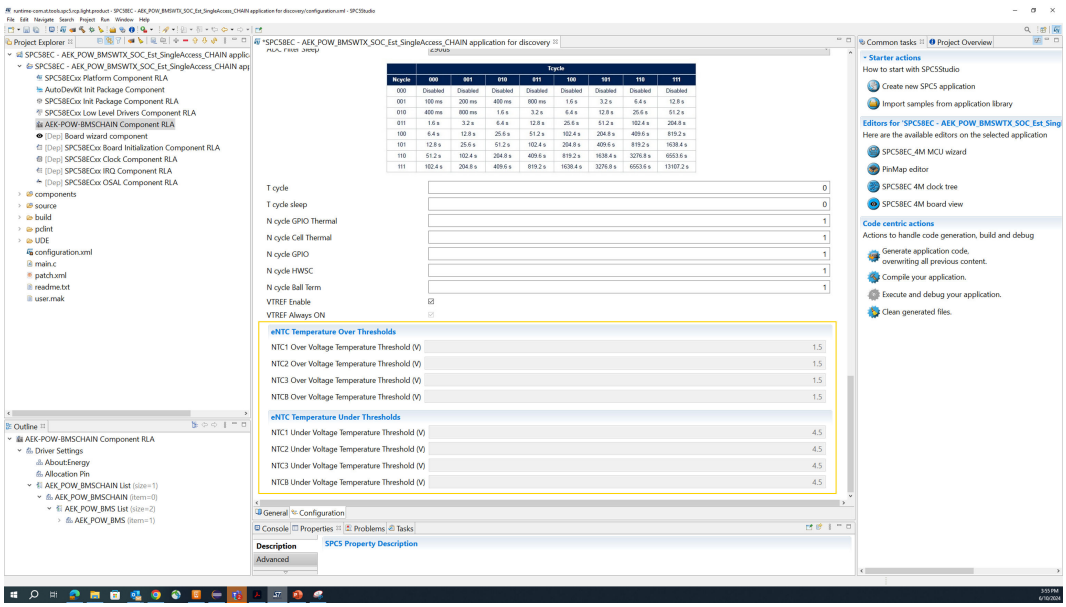
Step 25. Configure Vtref voltage regulator. If the “Vtref Enable” field is enabled, the voltage regulator will be active when required (for example, during voltage acquisition on GPIOs if the Ratiometric mode is enabled). If the “Vtref Always ON” field is enabled, the voltage regulator is always active.

Figure 46. Vtref voltage regulator configuration



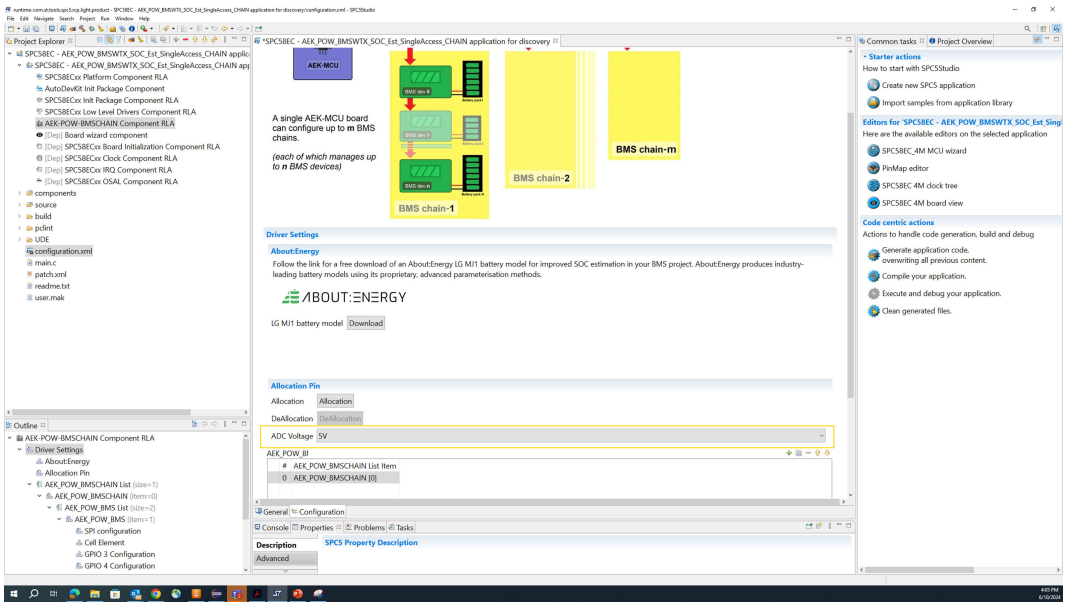
Step 26. Configure temperature thresholds (in voltage) if the first chain node is an AEK-POW-BMSNOTX evaluation board. If the first chain node is a different board, the thresholds are not customizable.

Figure 47. Temperature threshold configuration



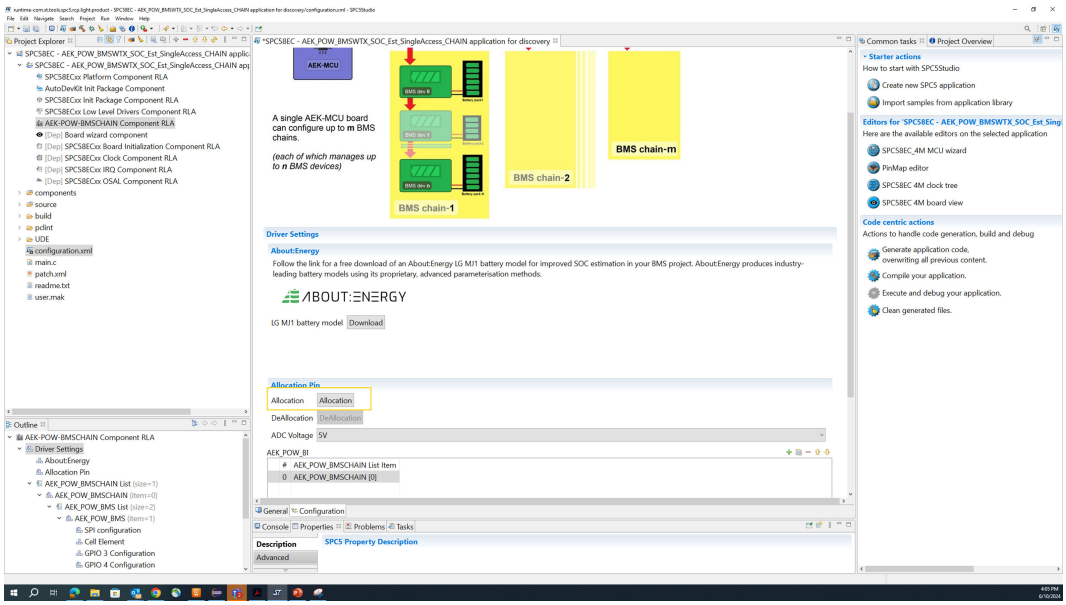
Step 27. In case an AEK-POW-BMSNOTX is included in a chain, you must configure the reference voltage of the MCU ADC (ADC Voltage field). This configuration does not replace the configuration of the MCU board jumpers.

Figure 48. MCU ADC voltage configuration when including an AEK-POW-BMSNOTX



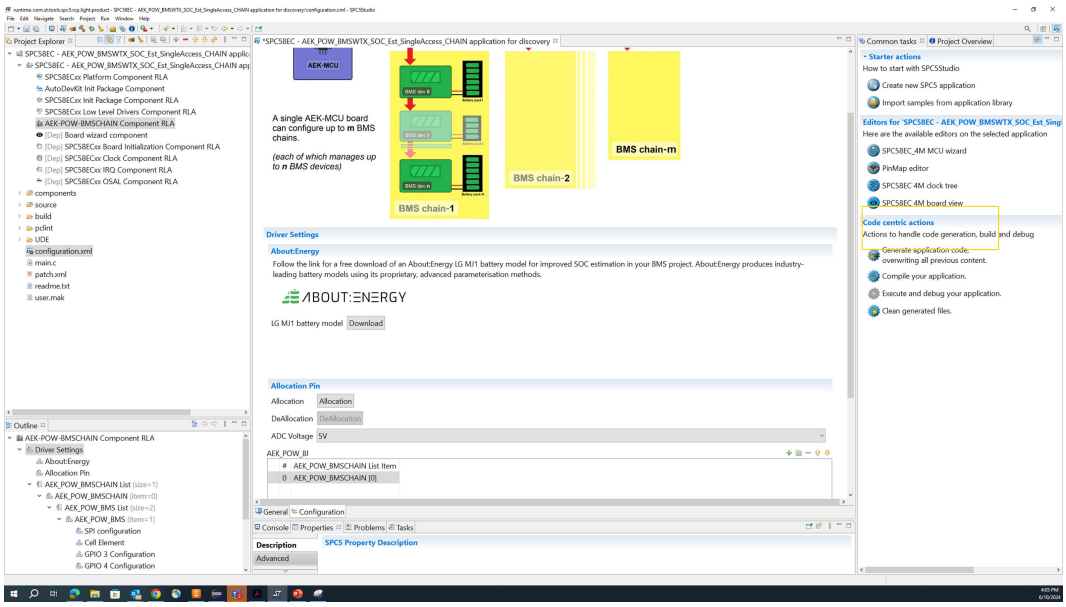
Step 28. After configuring all chain nodes, proceed with automatic allocation for all customized chains.

Figure 49. Automatic allocation for all customized chains



Step 29. Generate the AEK-POW-BMSCHAIN library C code and compile the project.

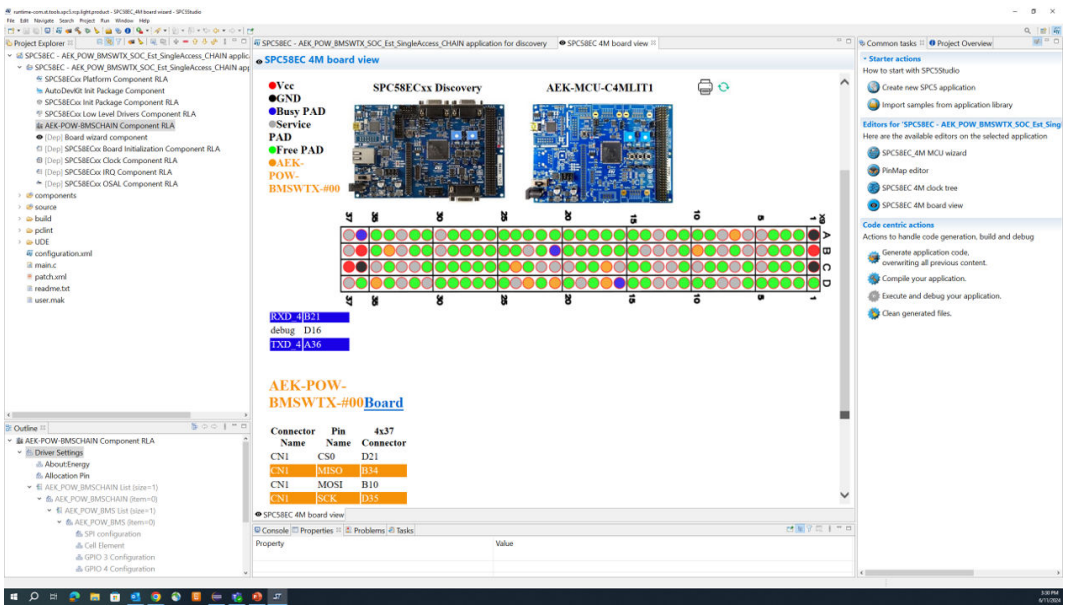
Figure 50. AEK-POW-BMSCHAIN library C code generation and compilation



Step 30. Include **AEK_POW_BMS63CHAIN_app_mng.h** in your main application and write your application code.

Step 31. Click on [Board View] to display the hardware connection between the AEK-MCU-C4MLIT1 board and the BMS chain.

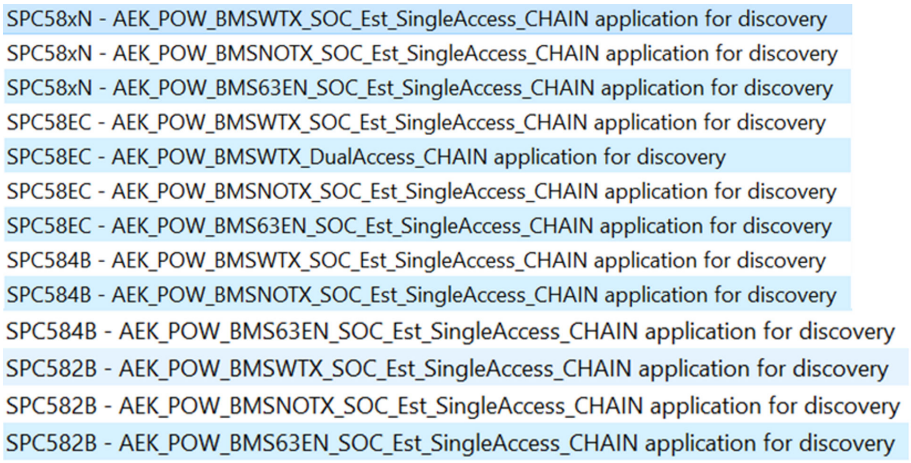
Figure 51. Board View



10.5 Available demos

In the AutoDevKit starting from release 2.4.0, there are several available demos for our BMS boards:

Figure 52. **Available demos**



All “SOC_Est_SingleAccess” demos can estimate SOC for a single node chain. (A demo has been created for each node topology).

The AEK_POW_BMSWTX_DualAccess_CHAIN shows how a chain with two nodes (AEK-POW-BMSWTX and AEK-POW-BMS63EN) works in dual access ring. This demo does not estimate the SOC.

Only demos for SPC58EC with SOC_Est_SingleAccess plot SOC, voltage and current via serial terminal.

10.5.1 Centralized application (SPC58EC - AEK_POW_BMSWTX_SOC_Est_SingleAccess_CHAIN application for discovery demo)

This application is based on the SPC58EC microcontroller and can estimate the SOC of 14 cells (in our case, the batteries used are INR 18650 MJ1 by LG) connected with the AEK-POW-BMSWTX evaluation board. The results of SOC estimation, cell voltage, NTC temperatures, and current are printed via a serial port with a speed rate of 115200 bps.

Below is the main code for the SPC58EC - AEK_POW_BMSWTX_SOC_Est_SingleAccess_CHAIN application for discovery demo:

Figure 53. **SPC58EC - AEK_POW_BMSWTX_SOC_Est_SingleAccess_CHAIN code**

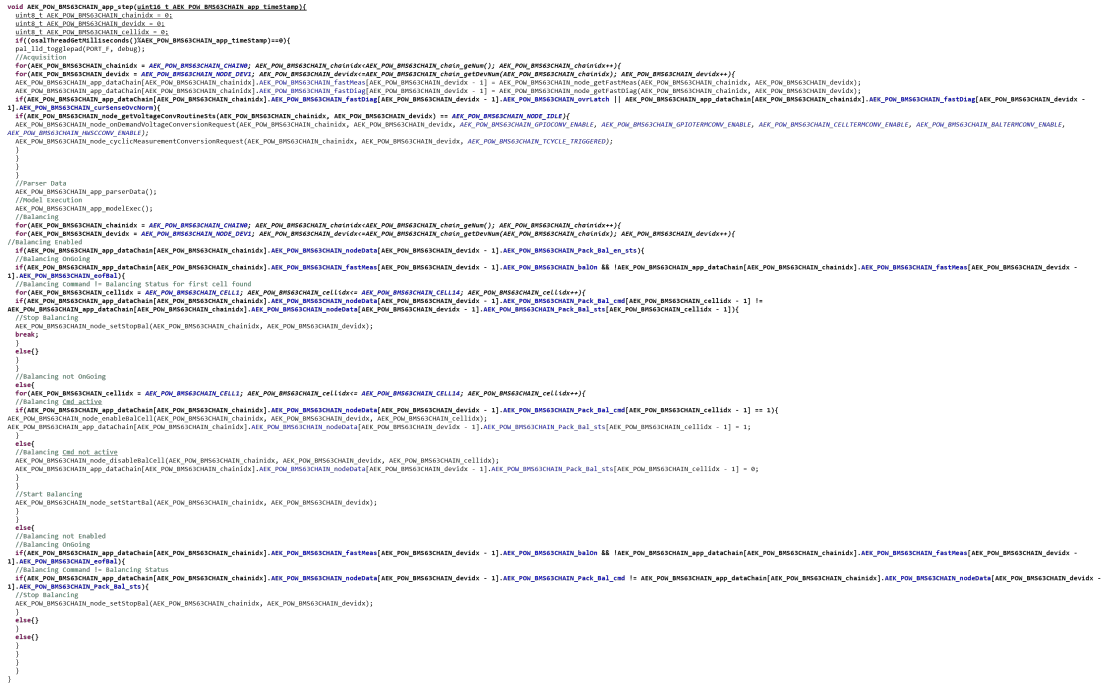
```
/**************************************************************************
*
* Copyright © 2015-2021 STMicroelectronics - All Rights Reserved
*
* This software is licensed under SLA0098 terms that can be found in the
* DM00779817_1_0.pdf file in the licenses directory of this software product.
*
* THIS SOFTWARE IS DISTRIBUTED "AS IS," AND ALL WARRANTIES ARE DISCLAIMED,
* INCLUDING MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
*
**************************************************************************/

/* Inclusion of the main header files of all the imported components in the
   order specified in the application wizard. The file is generated
   automatically.*/
#include "AEK_POW_BMS63CHAIN_app_mng.h"

#define AEK_POW_BMS63CHAIN_TIMESTEP 1000 //msec
#define AEK_POW_BMS63CHAIN_SERIAL_TIMESTEP 1000 //msec

void main_core0(void) {
  irqIsrEnable();
  AEK_POW_BMS63CHAIN_app_serialInit();
  for(;;){
  AEK_POW_BMS63CHAIN_app_serialStep(AEK_POW_BMS63CHAIN_SERIAL_TIMESTEP);
  }
}

int main(void) {
  componentsInit();
  irqIsrEnable();
  /* Application init */
  AEK_POW_BMS63CHAIN_app_init();
  runCore0();
  /* Application main loop.*/
  for ( ; ; ) {
  AEK_POW_BMS63CHAIN_app_step(AEK_POW_BMS63CHAIN_TIMESTEP);
  }
}
```

This application enables two cores:

- Core2 (main core) – Used to start the manager process (AEK_POW_BMS63CHAIN_app_step()).

This process is called every 1000ms.

Figure 54. Core 2 enablement



```
void AEK_POW_BMS63CHAIN_app_step(uint16_t AEK_POW_BMS63CHAIN_app_timeStamp){
  uint8_t AEK_POW_BMS63CHAIN_chainidx = 0;
  uint8_t AEK_POW_BMS63CHAIN_devidx = 0;
  uint8_t AEK_POW_BMS63CHAIN_cellidx = 0;
  if((osalThreadGetMilliseconds()%AEK_POW_BMS63CHAIN_app_timeStamp)==0){
  pal_lld_togglepad(PORT_F, debug);
  //Acquisition
  for(AEK_POW_BMS63CHAIN_chainidx = AEK_POW_BMS63CHAIN_CHAIN0; AEK_POW_BMS63CHAIN_chainidx<AEK_POW_BMS63CHAIN_chain_genNum(); AEK_POW_BMS63CHAIN_chainidx++){
  for(AEK_POW_BMS63CHAIN_devidx = AEK_POW_BMS63CHAIN_NODE_DEV1; AEK_POW_BMS63CHAIN_devidx<=AEK_POW_BMS63CHAIN_chain_getDevNum(AEK_POW_BMS63CHAIN_chainidx); AEK_POW_BMS63CHAIN_devidx++){
  AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_fastMeas[AEK_POW_BMS63CHAIN_devidx - 1] = AEK_POW_BMS63CHAIN_node_getFastMeas(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx);
  AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_fastDiag[AEK_POW_BMS63CHAIN_devidx - 1] = AEK_POW_BMS63CHAIN_node_getFastDiag(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx);
  if(AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_fastDiag[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_ovrLatch || AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_fastDiag[AEK_POW_BMS63CHAIN_devidx -
1].AEK_POW_BMS63CHAIN_curSenseOvcNorm){
  if(AEK_POW_BMS63CHAIN_node_getVoltageConvRoutineSts(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx) == AEK_POW_BMS63CHAIN_NODE_IDLE){
  AEK_POW_BMS63CHAIN_node_onDemandVoltageConversionRequest(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx, AEK_POW_BMS63CHAIN_GPIOCONV_ENABLE, AEK_POW_BMS63CHAIN_GPIOTERMCONV_ENABLE, AEK_POW_BMS63CHAIN_CELLTERMCONV_ENABLE, AEK_POW_BMS63CHAIN_BALTERMCONV_ENABLE,
AEK_POW_BMS63CHAIN_MASCCONV_ENABLE);
  AEK_POW_BMS63CHAIN_node_cyclicMeasurementConversionRequest(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx, AEK_POW_BMS63CHAIN_TCYCLE_TRIGGERED);
  }
  }
  }
  //Parser Data
  AEK_POW_BMS63CHAIN_app_parserData();
  //Model Execution
  AEK_POW_BMS63CHAIN_app_modelExec();
  //Balancing
  for(AEK_POW_BMS63CHAIN_chainidx = AEK_POW_BMS63CHAIN_CHAIN0; AEK_POW_BMS63CHAIN_chainidx<AEK_POW_BMS63CHAIN_chain_genNum(); AEK_POW_BMS63CHAIN_chainidx++){
  for(AEK_POW_BMS63CHAIN_devidx = AEK_POW_BMS63CHAIN_NODE_DEV1; AEK_POW_BMS63CHAIN_devidx<=AEK_POW_BMS63CHAIN_chain_getDevNum(AEK_POW_BMS63CHAIN_chainidx); AEK_POW_BMS63CHAIN_devidx++){
 //Balancing Enabled
  if(AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_nodeData[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_Pack_Bal_en_sts){
  //Balancing OnGoing
  if(AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_fastMeas[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_balOn && !AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_fastMeas[AEK_POW_BMS63CHAIN_devidx -
1].AEK_POW_BMS63CHAIN_eofBal){
  //Balancing Command != Balancing Status for first cell found
  for(AEK_POW_BMS63CHAIN_cellidx = AEK_POW_BMS63CHAIN_CELL1; AEK_POW_BMS63CHAIN_cellidx<= AEK_POW_BMS63CHAIN_CELL14; AEK_POW_BMS63CHAIN_cellidx++){
  if(AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_nodeData[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_Pack_Bal_cmd[AEK_POW_BMS63CHAIN_cellidx - 1] !=
AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_nodeData[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_Pack_Bal_sts[AEK_POW_BMS63CHAIN_cellidx - 1]){
  //Stop Balancing
  AEK_POW_BMS63CHAIN_node_setStopBal(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx);
  break;
  }
  else{}
  }
  }
  //Balancing not OnGoing
  else{
  for(AEK_POW_BMS63CHAIN_cellidx = AEK_POW_BMS63CHAIN_CELL1; AEK_POW_BMS63CHAIN_cellidx<= AEK_POW_BMS63CHAIN_CELL14; AEK_POW_BMS63CHAIN_cellidx++){
  //Balancing Cmd active
  if(AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_nodeData[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_Pack_Bal_cmd[AEK_POW_BMS63CHAIN_cellidx - 1] == 1){
AEK_POW_BMS63CHAIN_node_enableBalCell(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx, AEK_POW_BMS63CHAIN_cellidx);
AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_nodeData[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_Pack_Bal_sts[AEK_POW_BMS63CHAIN_cellidx - 1] = 1;
  }
  else{
  //Balancing Cmd not active
  AEK_POW_BMS63CHAIN_node_disableBalCell(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx, AEK_POW_BMS63CHAIN_cellidx);
  AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_nodeData[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_Pack_Bal_sts[AEK_POW_BMS63CHAIN_cellidx - 1] = 0;
  }
  }
  //Start Balancing
  AEK_POW_BMS63CHAIN_node_setStartBal(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx);
  }
  }
  else{
  //Balancing not Enabled
  //Balancing OnGoing
  if(AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_fastMeas[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_balOn && !AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_fastMeas[AEK_POW_BMS63CHAIN_devidx -
1].AEK_POW_BMS63CHAIN_eofBal){
  //Balancing Command != Balancing Status
  if(AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_nodeData[AEK_POW_BMS63CHAIN_devidx - 1].AEK_POW_BMS63CHAIN_Pack_Bal_cmd != AEK_POW_BMS63CHAIN_app_datachain[AEK_POW_BMS63CHAIN_chainidx].AEK_POW_BMS63CHAIN_nodeData[AEK_POW_BMS63CHAIN_devidx -
1].AEK_POW_BMS63CHAIN_Pack_Bal_sts){
  //Stop Balancing
  AEK_POW_BMS63CHAIN_node_setStopBal(AEK_POW_BMS63CHAIN_chainidx, AEK_POW_BMS63CHAIN_devidx);
  }
  else{}
  }
  else{}
  }
  }
  }
}
```

This process starts the cyclic conversion routine to measure cell voltage, cell current, and GPIO NTC temperature.

It also performs the state of charge estimation by using the C-code generated through a Simulink model and enables the manual balancing during the Normal state of the FSM.

In case of Fault condition activated, on-demand conversion routine will replace cyclic conversion routine.

- Core0 (auxiliary core) –
  Used to start a process to print via serial the following data (with a speed rate of 115200 bps):
  - State of charge percentage for each cell.
  - Cell voltage values.
  - Current value.
  - Balancing status.

This process is called every 1000 ms.

10.5.2 Dual ring application (SPC58EC - AEK_POW_BMSWTX_DualAccess_CHAIN application for the discovery demo)

This application is based on the SPC58EC microcontroller and can measure cell voltage, current and GPIO voltage for 2 AEK-POW-BMSWTX evaluation boards.

Find below the main code for the SPC58EC - AEK_POW_BMSWTX_DualAccess_CHAIN application for the discovery demo:

Figure 55. **SPC58EC - AEK_POW_BMSWTX_DualAccess_CHAIN code**

```
/*****************************************************************************
*
* Copyright © 2015-2021 STMicroelectronics - All Rights Reserved
*
* This software is licensed under SLA0098 terms that can be found in the
* DM00779817_1_0.pdf file in the licenses directory of this software product.
*
* THIS SOFTWARE IS DISTRIBUTED "AS IS," AND ALL WARRANTIES ARE DISCLAIMED,
* INCLUDING MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
*
*****************************************************************************/

/* Inclusion of the main header files of all the imported components in the
   order specified in the application wizard. The file is generated
   automatically.*/
#include "AEK_POW_BMS63CHAIN_app_mng.h"

#define AEK_POW_BMS63CHAIN_TIMESTEP 200 //msec
#define AEK_POW_BMS63CHAIN_SERIAL_TIMESTEP 1000 //msec

void main_core0(void) {
  irqIsrEnable();
  AEK_POW_BMS63CHAIN_app_serialInit();
  for(;;){
  AEK_POW_BMS63CHAIN_app_serialStep(AEK_POW_BMS63CHAIN_SERIAL_TIMESTEP);
  }
}

int main(void) {
  componentsInit();
  irqIsrEnable();
  /* Application init */
pal_lld_setpad(PORT_F, debug);
  AEK_POW_BMS63CHAIN_app_init();

  runCore0();
  /* Application main loop.*/
  for ( ; ; ) {
  AEK_POW_BMS63CHAIN_app_step(AEK_POW_BMS63CHAIN_TIMESTEP);
  }
}
```

This application is based on the SPC58EC microcontroller and can measure cell voltage, current and GPIO voltage for 2 AEK-POW-BMSWTX evaluation boards.

Find below the main code for the SPC58EC - AEK_POW_BMSWTX_DualAccess_CHAIN application for the discovery demo:

Figure 56. **SPC58EC - AEK_POW_BMSWTX_DualAccess_CHAIN code**

```
/*****************************************************************************
*
* Copyright © 2015-2021 STMicroelectronics - All Rights Reserved
*
* This software is licensed under SLA0098 terms that can be found in the
* DM00779817_1_0.pdf file in the licenses directory of this software product.
*
* THIS SOFTWARE IS DISTRIBUTED "AS IS," AND ALL WARRANTIES ARE DISCLAIMED,
* INCLUDING MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
*
*****************************************************************************/

/* Inclusion of the main header files of all the imported components in the
   order specified in the application wizard. The file is generated
   automatically.*/
#include "AEK_POW_BMS63CHAIN_app_mng.h"

#define AEK_POW_BMS63CHAIN_TIMESTEP 200 //msec
#define AEK_POW_BMS63CHAIN_SERIAL_TIMESTEP 1000 //msec

void main_core0(void) {
  irqIsrEnable();
  AEK_POW_BMS63CHAIN_app_serialInit();
  for(;;){
  AEK_POW_BMS63CHAIN_app_serialStep(AEK_POW_BMS63CHAIN_SERIAL_TIMESTEP);
  }
}

int main(void) {
  componentsInit();
  irqIsrEnable();
  /* Application init */
pal_lld_setpad(PORT_F, debug);
  AEK_POW_BMS63CHAIN_app_init();

  runCore0();
  /* Application main loop.*/
  for ( ; ; ) {
  AEK_POW_BMS63CHAIN_app_step(AEK_POW_BMS63CHAIN_TIMESTEP);
  }
}
```

10.5.3 Single centralized application (SPC58EC – AEK_POW_BMS63EN_SOC_Estimation_Single application for discovery demo)

This application is based on the SPC58EC microcontroller and can estimate the SOC of 14 cells in a single BMS node (in our case, the battery used are INR 18650 MJ1 by LG) connected with an AEK-POW-BMS63EN evaluation board.

The results of SOC estimation, cell voltage, NTC temperatures, and current are printed via serial port with a speed rate of 115200 bps.

Find below the main code for the SPC58EC – AEK_POW_BMS63EN_SOC_Estimation_Single application for discovery demo:

```
*
 * Copyright Â© 2022 STMicroelectronics - All Rights Reserved
 *
 * License terms: STMicroelectronics Proprietary in accordance with licensing
 * terms SLA0089 at www.st.com.
 *
 * THIS SOFTWARE IS DISTRIBUTED "AS IS," AND ALL WARRANTIES ARE DISCLAIMED,
 * INCLUDING MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
 *
 * EVALUATION ONLY â€" NOT FOR USE IN PRODUCTION
 *****************************************************************************/
/* Inclusion of the main header files of all the imported components in the
 order specified in the application wizard. The file is generated
 automatically.*/
```

```
#include "components.h"
#include "AEK_POW_BMS63EN_Appl.h"
#include "wkpu_lld_cfg.h"
#include "serial_lld_cfg.h"
#include <stdio.h>
#include <string.h>
volatile uint8_t i_device_disp=0;
void main_core0(void) {
  char message[11];
  AEK_POW_BMS63EN_module_t AEK_POW_BMS63EN_BatteryModule;
  uint8_t cell_idx = 0;
  /* Enable Interrupts */
  irqIsrEnable();
  sd_lld_start(&SD5,&serial_config_BMS_serial);
  /* Application main loop.*/
  for ( ; ; ) {
  if((osalThreadGetMilliseconds()%100)==0){
    AEK_POW_BMS63EN_BatteryModule = AEK_POW_BMS63EN_GetModule(i_device_disp);
//SOC Elaboration Done
sprintf(message, "DEV:%d     \n", (int)(i_device_disp+1));
sd_lld_write(&SD5, (uint8_t *)message, (uint16_t)(sizeof(message)/sizeof(message[0])));
//Printing SOC
for(cell_idx = AEK_POW_BMS63EN_CELL1; cell_idx <= AEK_POW_BMS63EN_CELL14; cell_idx ++){
sprintf(message, "S%.2d:%.3d    ", cell_idx+1, (int)((AEK_POW_BMS63EN_BatteryModule.AEK_POW_B
MS63EN_Pack_SOC[cell_idx]) * 100));
sd_lld_write(&SD5, (uint8_t *)message, (uint16_t)(sizeof(message)/sizeof(message[0])));
}
sprintf(message,"          \n");
sd_lld_write(&SD5, (uint8_t *)message, (uint16_t)(sizeof(message)/sizeof(message[0])));
//Printing Bal
for(cell_idx = AEK_POW_BMS63EN_CELL1; cell_idx <= AEK_POW_BMS63EN_CELL14; cell_idx ++){
sprintf(message, "B%.2d:%.3d    ", cell_idx+1, AEK_POW_BMS63EN_BatteryModule.AEK_POW_BMS63EN_
Pack_Bal_cmd[cell_idx]);
sd_lld_write(&SD5, (uint8_t *)message, (uint16_t)(sizeof(message)/sizeof(message[0])));
}
sprintf(message,"          \n");
sd_lld_write(&SD5, (uint8_t *)message, (uint16_t)(sizeof(message)/sizeof(message[0])));
//Printing Voltage
for(cell_idx = AEK_POW_BMS63EN_CELL1; cell_idx <= AEK_POW_BMS63EN_CELL14; cell_idx ++){
sprintf(message, "V%.2d:%.3f  ", cell_idx+1, AEK_POW_BMS63EN_BatteryModule.AEK_POW_BMS63EN_Pa
ck_CellVoltage[cell_idx]);
sd_lld_write(&SD5, (uint8_t *)message, (uint16_t)(sizeof(message)/sizeof(message[0])));
}
sprintf(message,"          \n");
sd_lld_write(&SD5, (uint8_t *)message, (uint16_t)(sizeof(message)/sizeof(message[0])));
//Printing Current
sprintf(message, "C:%.4f    \n", AEK_POW_BMS63EN_BatteryModule.AEK_POW_BMS63EN_Pack_Current);
sd_lld_write(&SD5, (uint8_t *)message, (uint16_t)(sizeof(message)/sizeof(message[0])));
}
  }
}
void *sbrk(size_t incr)
{
extern uint8_t __heap_base__;
extern uint8_t __heap_end__;
static uint8_t *p=&__heap_base__;
static uint8_t *newp;
newp = p+ incr;
if(newp> &__heap_end__)
{
return (void*)-1;
}
return p =newp;
}
int main(void) {
componentsInit();
irqIsrEnable();
wkpu_lld_start(&WKPUD1, &wkpu_config_wkpu_cfg);
AEK_POW_BMS63EN_init();
runCore0();
```

```
AEK_POW_BMS63EN_Start_Mgn_exec();
while (1) {
}
}
```

This application enables two cores:

- Core2 (main core)
  - Used to start the manager process (AEK_POW_BMS63EN_Mng_exec()) through a PIT (i.e., a 5 Hz programmable timer:)

```
void AEK_POW_BMS63EN_Mgn_exec(){
pal_lld_togglepad(PORT_F,LED3);
AEK_POW_BMS63EN_soc_elab_sts = AEK_POW_BMS63EN_SOC_ELABORATION_BUSY;
AEK_POW_BMS63EN_VA_Measurement();
AEK_POW_BMS63EN_TEMP_Measurement();
if(AEK_COM_ISOSPI_GetFault(ISOSPI_DEV0)){
AEK_POW_BMS63EN_DIAG_Measurement();
}
AEK_POW_BMS63EN_Model_exec();
AEK_POW_BMS63EN_Balancing();
AEK_POW_BMS63EN_soc_elab_sts = AEK_POW_BMS63EN_SOC_ELABORATION_DONE;
pal_lld_togglepad(PORT_F,LED3);
}
```

    This process starts the manual conversion routine to measure the cell voltage, current, and GPIO NTC temperature.
    Moreover, this process performs the SOC estimation by using the C-code generated through a Simulink model and enables the manual balancing during the Normal state of the FSM.
- Core0 (auxiliary core)
  - Used to start a process that can print via serial the following data (with a speed rate of 115200 bps):
    - State of charge percentage for each cell
    - Cell voltage values
    - Current value
    - Balancing status
  - This process is called every 100 ms.

11 Available APIs

Table 1. Chain management APIs

API name	Descripition
AEK_POWBMS63CHAIN_chain_init	Initializes the BMS chain
AEK_POWBMS63CHAIN_chain_wakeup	Init BMS chain wakeup
AEK_POW_BMS63CHAIN_chain_addressingProcedure	Init BMS chain addressing procedure
AEK_POW_BMS63CHAIN_chain_parSetting	Set BMS chain parameters
AEK_POW_BMS63CHAIN_chain_getDevNum	Get the Node number of the BMS chain
AEK_POW_BMS63CHAIN_chain_geNum	Ge the BMS chain number.

Table 2. CRC APIs

API name	Descripition
AEK_POW_BMS63CHAIN_crc_elab	Get the SPI frame with CRC.

Table 3. ISOSPI management APIs

API name	Descripition
AEK_POW_BMS63CHAIN_iso_init	Init ISOSPI for BMSCHAIN.
AEK_POW_BMS63CHAIN_iso_signalSetting	Init ISOSPI for BMSCHAIN.
AEK_POW_BMS63CHAIN_iso_setDis	Setting Disable Pin of ISOSPI.
AEK_POW_BMS63CHAIN_iso_setISOFreq	Setting ISOSPI Frequency.
AEK_POW_BMS63CHAIN_iso_setTXEn	Setting TXEn Pin of ISOFreq.
AEK_POW_BMS63CHAIN_iso_setTxAmp	Setting ISOSPI Trasmission Amplitude.

Table 4. Node management APIs

API name	Descripition
AEK_POW_BMS63CHAIN_node_getTwoComp	Get 2' Complement from spi data read.
AEK_POW_BMS63CHAIN_node_getInverseTwoComp	Get 2' Inverse Complement from spi data read.
AEK_POW_BMS63CHAIN_node_enableCell	Enable Cell of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_disableCell	Disable Cell of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltageCell	Disable Cell of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltagePackSum	Get Pack Sum Voltage value of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltagePack	Get Pack Voltage value of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltageTRef	Get Voltage Reference value of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getCalibCurrent	Get Calib Current value of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getInstCurrent	Get Inst Current value of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getChipTemperature	Get Chip Temperature value of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getOTChipTemperature	Get OverTemperature Chip flag of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getGPIOVoltage	Get GPIO Voltage value of Node of BMS Chain.

API name	Description
AEK_POW_BMS63CHAIN_node_setGPIOUTT	Set GPIO UnderTemperature Threshold of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setGPIOOTT	Set GPIO OverTemperature Threshold of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setGPIORatioAbsMode	Set GPIO Measurement Mode of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setGPIOConf	Set GPIO Configuration of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setGPIOMsk	Set GPIO Fault Mask of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setOVCSleepMsk	Set OverCurrent Fault Mask Sleep status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setOVCNormMsk	Set OverCurrent Fault Mask Normal status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setGPIO7WUEn	Enable Wakeup from GPIO7 of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setGPIO7WUDis	Disable Wakeup from GPIO7 of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setVCellOVT	Set Cell OverVoltage Cell Threshold of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setVCellUVT	Set Cell UnderVoltage Cell Threshold of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setVBattSumOVT	Set Battery Sum OverVoltage Cell Threshold of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setVBattSumUVT	Set Battery Sum UnderVoltage Cell Threshold of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setCSAThNorm	Set CSA Threshold Normal status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setCSAThSleep	Set CSA Threshold Sleep status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltageConvRoutineSts	Get FSM of Voltage Conversion routine status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setSOC	Set Request SOC routine (on-demand Conversion) of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setCyclicConf	Set Request Cyclic Conversion routine of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setCylicMode	Set Cyclic Mode of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_onDemandVoltageConversionRequest	Request OnDemand Conversion Routine of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_cyclicMeasurementConversionRequest	Request Cyclic Conversion Routine of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_cyclicDiagnosticConversionRequest	Request Cyclic Diagnostic Conversion Routine of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setGPIOConv	Set GPIO Conv flag of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setGPIOTermConv	Set GPIO Term Conv flag of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setCellTermConv	Set Cell Term Conv flag of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setBalTermConv	Set Balancing Term Conv flag of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setHWSCConv	Set Hardware Self Check Conv flag of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setCoulombCounter	Set Coulomb Counter flag of Node of BMS Chain.

API name	Description
AEK_POW_BMS63CHAIN_node_setCommTimeout	Set Communication Timeout flag of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setCyclicUpdate	Set Cyclic Update flag of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setNcylceGPIOTerm	Set NCycle GPIO Term of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setNcylceCellTerm	Set NCycle Cell Term of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setNcylceBalTerm	Set NCycle Balancing Term of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setNcylceGPIO	Set NCycle GPIO of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setNcylceHWSC	Set NCycle Hardware Self Check of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setADCFilterCycle	Set ADC Filter Cycle of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setADCFilterSleep	Set ADC Filter Sleep of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setADCFilterSOC	Set ADC Filter SOC of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setTCycle	Set TCycle of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setTCycleSleep	Set TCycle Sleep of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setVTrefEnable	Set VTref Enable/Disable of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setVTrefAlwaysON	Set VTref Always ON Enable/Disable of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getFSMsts	Get Voltage Convertion Routine status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setSilentBalancingMode	Set Silent Balancing of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setSWReset	Request Software Reset of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_go2Sleep	Request Go To Sleep transition of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setISOTXH	Set ISOTX High of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_wakeup	Request Wakeup of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setISOFreqSel	Set ISOFreqSel of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setOutResTx	Set OUTResTX of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setChipID	Set CHIPID of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getChipID	Get CHIPID of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setFarthestUnit	Set FarthestUnit of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_lockISOHISOFreq	Set LockISOFreq of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_unLockISOHISOFreq	UnLock ISOFreq of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getSPIError	Get SPI Communication Error of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getFrameSPIError	Get Frame Error of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getLossAGnd	Get AGndLoss Error of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getLossDGnd	Get DGndLoss Error of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getLossCGnd	Get CGndLoss Error of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getLossGndRef	Get Ref Gnd Loss Error of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getTrimmCalOK	Get TrimCalOK of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getClockMonInitDone	Get Clock Monitor Init Done of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getOSCFail	Get Oscillator Error of Node of BMS Chain.

API name	Description
AEK_POW_BMS63CHAIN_node_getSenseMinusOpen	Get Sense Minus Open Error of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getSensePlusOpen	Get Sense Plus Open Error of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getCurrSenseOVCNorm	Get OverCurrent Sense Normal status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getCurrSenseOvcSleep	Get OverCurrent Sense Normal status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getFaultLineLSts	Get Fault L status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getFaultHLineFault	Get Fault H status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getCommTimeoutFlt	Get Communication Timeout Filter of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getEOFBal	Get End Of Balancing status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getBalON	Get Balancing ON status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getTimedBalAcc	Get Timed Balancing Acc status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getEoBTimeError	Get End Of Balancing Time error status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getHeartBeatEn	Get HeartBeat Enable status of Node of BMS Chain
AEK_POW_BMS63CHAIN_node_getHeartBeatFault	Get HeartBeat Fault status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getCoulombCounterEn	Get Coulomb Counter Enable status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getCoCouOvf	Get Coulomb Counter Overflow status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getOverLatch	Get Overlatch status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getDutyON	Get duty ON of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getConfCyclicEN	Get Cylic EN status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getTCyclicOVF	Get Tcylic Overflow status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getEEPROMDownloadDone	Get EEPROM Download Done status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getEEPROMCRCErrCalRam	Get EEPROM CRC Cal RAM status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getEEPROMCRCErrSect0	Get EEPROM CRC Error Sect0 status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getRAMCRCErr	Get RAM CRC Error status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getWUgpio7	Get Wakeup from GPIO7 status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getWUspi	Get Wakeup from SPI status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getWUISOLine	Get Wakeup from ISOLINE status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getWUFaultH	Get Wakeup from FaultH status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getWUCyc	Get Wakeup from Cylic Mode status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltageCellOV	Get OverVoltage Cell status of Node of BMS Chain.

API name	Description
AEK_POW_BMS63CHAIN_node_getVoltageCellUV	Get UnderVoltage Cell status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getCellOpen	Get Open Cell status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getBalOpen	Get Balancing Open Cell status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltagePackWarningUV	Get UnderVoltage Pack Warning status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltagePackCritUV	Get UnderVoltage Pack Critical status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltagePackCritOV	Get OverflowVoltage Pack Critical status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltagePackSumUV	Get UnderVoltage Pack Sum status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltagePackSumOV	Get OverVoltage Pack Sum status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltageBalUV	Get UnderVoltage Balancing status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVoltageBattCompBistFail	Get Voltage Batterty Comp Bist Fail status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getMuxBistFail	Get Mux Bist Fail status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getOpenBistFail	Get Open Bist Fail status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getGPIOBistFail	Get GPIO Bist Fail status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVTrefBistFail	Get VTRef Bist Fail status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getVBattDivBistFail	Get VBattDiv Bist Fail status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getFastMeas	Get FastMeasure struct of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getFastDiag	Get FastDiag struct of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setBalMode	Set Balancing Mode of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setStartBal	Set Balancing Start of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setStopBal	Set Balancing Stop of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_enableBalCell	Enable Balancing Cell of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_disableBalCell	Disable Balancing Cell of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_getBalSts	Get Balancing Status of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setThrTimedBalCell	Set Threshold Timed Balancing Cell of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setSleepBalConf	Set Sleep Balancing Configuration of Node of BMS Chain.
AEK_POW_BMS63CHAIN_node_setTimedBalRes	Set Timed Balancing Res of Node of BMS Chain.

Table 5. NTC management APIs

API name	Description
AEK_POW_BMS63CHAIN_ntc_init	NTC init for BMS Chain.
AEK_POW_BMS63CHAIN_ntc_deInit	NTC deinit for BMS Chain.
AEK_POW_BMS63CHAIN_ntc_getNTC1VoltageMeas	Get NTC1 voltage measure for BMS Chain.
AEK_POW_BMS63CHAIN_ntc_getNTC2VoltageMeas	Get NTC2 voltage measure for BMS Chain.
AEK_POW_BMS63CHAIN_ntc_getNTC3VoltageMeas	Get NTC3 voltage measure for BMS Chain.

API name	Description
AEK_POW_BMS63CHAIN_ntc_getNTCbVoltageMeas	Get NTCb voltage measure for BMS Chain.
AEK_POW_BMS63CHAIN_ntc_getNTC1Meas	Get NTC1 resistor measure for BMS Chain.
AEK_POW_BMS63CHAIN_ntc_getNTC2Meas	Get NTC2 resistor measure for BMS Chain.
AEK_POW_BMS63CHAIN_ntc_getNTC3Meas	Get NTC3 resistor measure for BMS Chain.
AEK_POW_BMS63CHAIN_ntc_getNTCbMeas	Get NTCb resistor measure for BMS Chain.

Table 6. SPI handling APIs

API name	Description
AEK_POW_BMS63CHAIN_spi_init	SPI protocol Init for BMS Chain.
AEK_POW_BMS63CHAIN_spi_deInit	SPI protocol Deinit for BMS Chain.
AEK_POW_BMS63CHAIN_spi_readReg	Get register value for BMS Chain.
AEK_POW_BMS63CHAIN_spi_readRegBurst	Send SPI burst command for BMS Chain.
AEK_POW_BMS63CHAIN_spi_writeReg	Send SPI write command for BMS Chain.

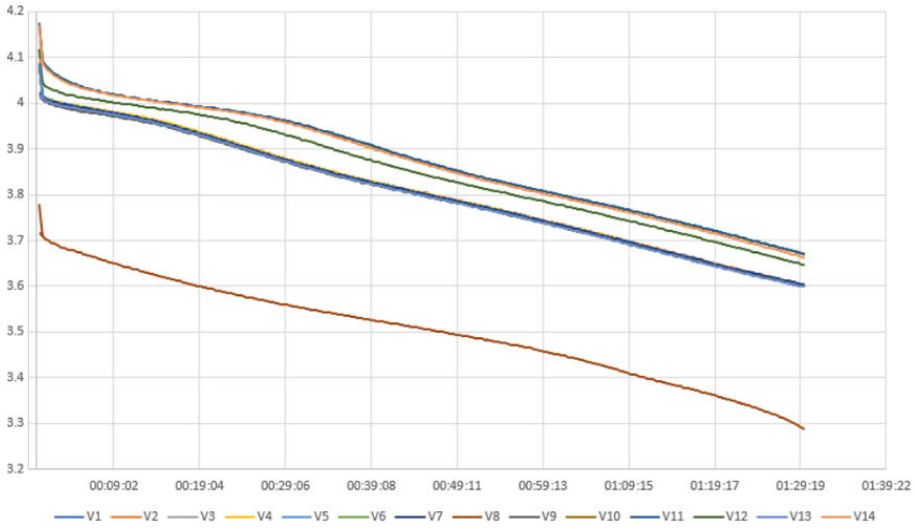
For further details on the APIs, refer to the doxygen documentation under the doc folder (see AEK-POW-BMSCHAIN component folder structure).

12 Waveforms

The following waveform shows the time trend of the voltages of the 14 cells connected to an AEK-POW-BMS63EN in a single centralized configuration, with an active load of 1 A and balancing activated by the cell 8 voltage value.

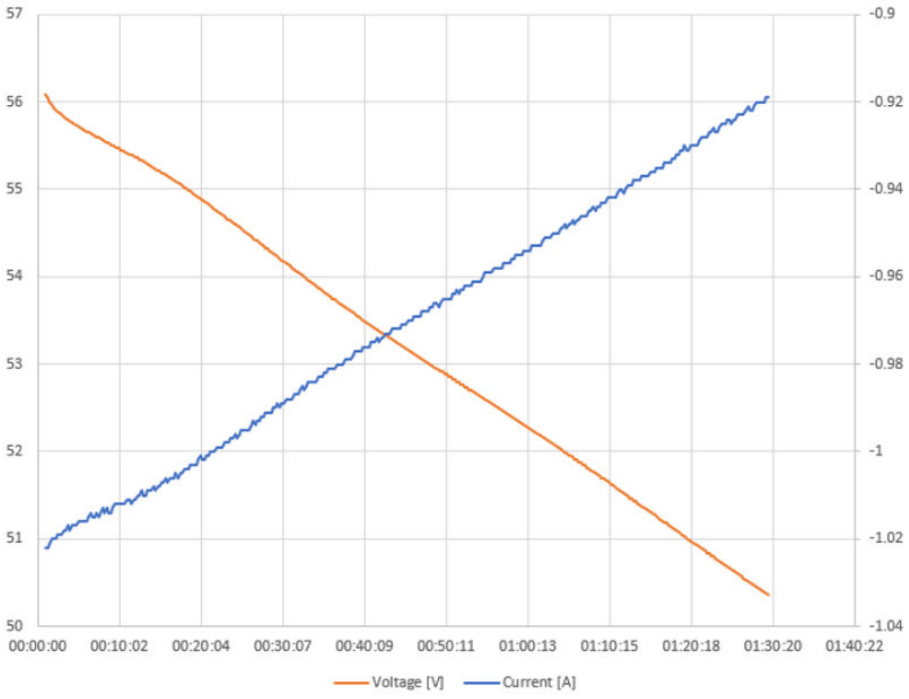
The x axis shows the time (in hours). The y axis shows the 14 cell voltages in Volts.

Figure 57. 14-cell voltage discharge with an active load of 1 A



The following waveform shows the time trend of the battery pack voltage (defined by 14 cells connected to an AEK-POW-BMS63EN in a single centralized configuration) with an active load of 1 A. The x axis shows the time (in hours). The y axis shows the battery pack voltage in Volts and the load current.

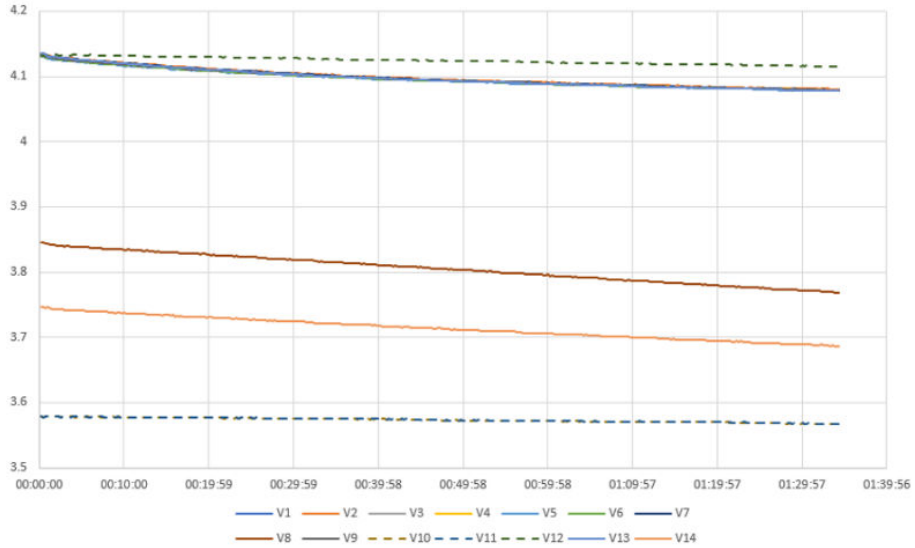
Figure 58. Battery pack voltage discharge with an active load of 1 A



The following waveform shows the time trend of the voltages of the 14 cells connected to an AEK-POW-BMS63EN in a single centralized configuration, when balancing is active.

The x axis shows the time (in hours). The y axis shows the voltages of the 14 cells in Volts.

Figure 59. 14 cell voltages during balancing



12.1 Process timing

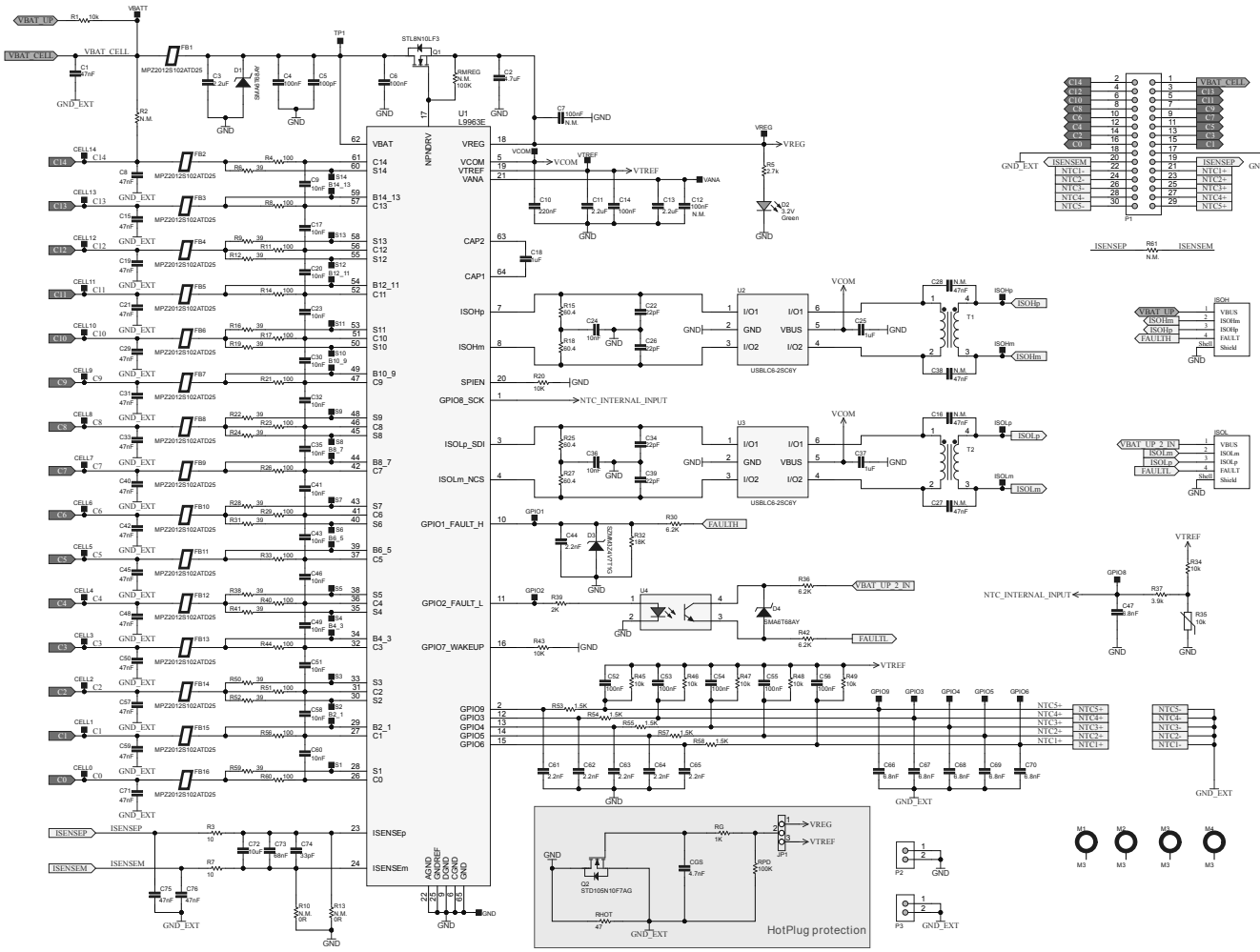
The following table defines the time length of the tasks defined in the SPC58EC - AEK_POW_BMSWTX_SOC_Est_SingleAccess_CHAIN application for discovery demo.

Table 7. Task process timing

Task	Timing
Voltage and current estimation	7 ms in cyclic conversion routine
SOC estimation	15.87 ms

13 AEK-POW-BMS63EN schematic diagrams

Figure 60. **AEK-POW-BMS63EN schematic diagram**



14 AEK-POW-BMS63EN bill of materials

Table 8. AEK-POW-BMS63EN bill of materials

Item	Q.ty	Ref.	Part / Value	Description	Manufacturer	Order code
1	18	C1, C8, C15, C19, C21, C29, C31, C33, C40, C42, C45, C48, C50, C57, C59, C71, C75, C76	47nF	0603 - 50V - X7R Class II	WE	885012206093
2	1	C2	4.7uF	1206 - 50V - X7R Class II	WE	885012208094
3	1	C3	2.2uF	1210 - 100V - X7R Class II	WE	885012209071
4	3	C4, C6, C14	100nF	0603 - 100V - X7R Class II	WE	885012206120
5	1	C5	100pF	0603 - 100V - X7R Class II	WE	885012206102
6	2	C7, C12	N.M.	0603	N.M.	N.M.
7	16	C9, C17, C20, C23, C24, C30, C32, C35, C36, C41, C43, C46, C49, C51, C58, C60	10nF	0603 - 50V - X7R Class II	WE	885012206089
8	1	C10	220nF	0603 - 50V - X7R Class II	WE	885012206125
9	2	C11, C13	2.2uF	0805 - 25V - X7R Class II	WE	885012207079
10	4	C16, C27, C28, C38	N.M.	1206	N.M.	N.M.
11	3	C18, C25, C37	1uF	0805 - 50V - X7R Class II	WE	885012207103
12	4	C22, C26, C34, C39	22pF	0603 - 50V - NP0 Class I	WE	885012006053
13	6	C44, C61, C62, C63, C64, C65	2.2nF	0603 - 50V - X7R Class II	WE	885012206085
14	6	C47, C66, C67, C68, C69, C70	6.8nF	0603 - 50V - X7R Class II	WE	885012206088
15	5	C52, C53, C54, C55, C56	100nF	0603 - 50V - X7R Class II	WE	885012206095
16	1	C72	10uF	1210 - 50V - X7R Class II	WE	885012209073
17	1	C73	68nF	0603 - 50V - X7R Class II	WE	885012206094
18	1	C74	33pF	0603 - 50V - NP0 Class I	WE	885012006054
19	1	CGS	4.7nF	0603 - 50V - X7R Class II	WE	885012206087
20	2	D1, D4	SMA6T68AY	Automotive 600 W, 68V TVS in SMA	STMicroelectronics	SMA6T68AY
21	1	D2	Green	0805 - Led Green - 3.2V	WE	150080GS75000
22	1	D3	SZMM3Z4V7T1G	4.7V Zener Voltage Regulators, 300mW	Onsemi	SZMM3Z4V7T1G

Item	Q.ty	Ref.	Part / Value	Description	Manufacturer	Order code
23	16	FB1, FB2, FB3, FB4, FB5, FB6, FB7, FB8, FB9, FB10, FB11, FB12, FB13, FB14, FB15, FB16	1K@100MHz	Ferrite Beads Multi-Layer Power 1KOhm 25% 100MHz 1.5A 0.15Ohm DCR 0805	TDK	MPZ2012S102ATD25
24	2	ISOH, ISOL	61400416021	USB 2.0 Type A, Receptacle, Horizontal, THT	WE	61400416021
25	1	JP1		THT Vertical 3 pins Header, Pitch 2.54 mm, Single Row	WE	61300311121
26	1	P1		2.00mm - WR-WTB - Male Dual Row Horizontal Shrouded Header w. positive locking	WE	62403021722
27	2	P2, P3	61300211121	2.54mm - WR-PHD Pin Header, THT, pitch 2.54mm, Single Row, Vertical, 2p	WE	61300211121
28	1	Q1	STL8N10LF3, PowerFLAT 5x6 WF	Automotive-grade N-channel 100 V, 25 mΩ typ., 7.8 A STripFET™ F3 Power MOSFET in a PowerFLAT™ 5x6 package	STMicroelectronics	STL8N10LF3
29	1	Q2	STD105N10F7AG, DPAK	Automotive-grade N-channel 100 V, 6.8 mΩ typ., 80 A, STripFET™ F7 Power MOSFET in a DPAK package	STMicroelectronics	STD105N10F7AG
30	1	R1	10k	1206 - ±1% - 0.66W	Panasonic	ERJUP8F1002V
31	1	R2	N.M.	0805	N.M.	N.M.
32	2	R3, R7	10	0603 - ±1% - 0.25W	Panasonic	ERJPA3F10R0V
33	15	R4, R8, R11, R14, R17, R21, R23, R26, R29, R33, R40, R44, R51, R56, R60	100	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1000V
34	1	R5	2.7k	0603 - ±1% - 0.125W	Vishay	MCT06030C2701FP500
35	14	R6, R9, R12, R16, R19, R22, R24, R28, R31, R38, R41, R50, R52, R59	39	2010 - ±1% - 1.25W	TE Connectivity	CRGP2010F39R
36	3	R10, R13, RMREG	N.M.	0603	N.M.	N.M.
37	4	R15, R18, R25, R27	60.4	0603 - ±1% - 0.1W	Panasonic	ERJ3EKF60R4V
38	2	R20, R43	10K	0603 - ±1% - 0.2W	Panasonic	ERJP03F1002V
39	3	R30, R36, R42	6.2K	0805 - ±1% - 0.5W	Panasonic	ERJP06F6201V
40	1	R32	18K	0603 - ±1% - 0.2W	Panasonic	ERJP03F1802V

Item	Q.ty	Ref.	Part / Value	Description	Manufacturer	Order code
41	6	R34, R45, R46, R47, R48, R49	10k	0805 - ±1% - 0.5W	Panasonic	ERJP6WF1002V
42	1	R35	10k	0603 - ±1% - 0.1W	TDK	NTCG163JH103HTDS
43	1	R37	3.9k	0603 - ±1% - 0.1W	Panasonic	ERJ3EKF3901V
44	1	R39	2K	0603 - ±1% - 0.2W	Panasonic	ERJP03F2001V
45	5	R53, R54, R55, R57, R58	1.5K	2010 - ±1% - 2W	TE Connectivity	35021K5FT
46	1	R61	N.M.	N.M.	N.M.	N.M.
47	1	RG	1K	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1001V
48	1	RHOT	47	2512 - ±5% - 1W	TE Connectivity	352047RJT
49	1	RPD	100K	0603 - ±1% - 0.25W	Panasonic	ERJP03F1003V
50	2	T1, T2	125uH	Transformer for BMS	WE	74941000
51	1	U1	L9963E, TQFP 64 10x10x1.0	Automotive chip for battery management applications with daisy chain up to 31 devices	STMicroelectronics	L9963E
52	2	U2, U3	USBLC6-2SC6Y, SOT23-6L	Automotive ESD protection for high speed interfaces.	STMicroelectronics	USBLC6-2SC6Y
53	1	U4	PS2703-1-A	4-Pin Phototransistor Optocoupler	Renesas	PS2703-1-A
54	1	for blister	60900213421	WR-PHD 2.54 mm Multi-Jumper Jumper with Test Point	WE	60900213421
55	4	for blister	970080365	WA-SPAII Plastic Spacer Stud, metric, internal/ internal	WE	970080365
56	4	for blister	97790403111	WA-SCRW Pan Head Screw w. cross slot M3	WE	97790403111
57	1	for blister	624030213322	WR-WTB 2.00 mm Female Dual Row Terminal Housing w. positive locking	WE	624030213322
58	30	for blister	62400113722	WR-WTB 2.00 mm Female Dual Row Crimp Contact	WE	62400113722

15 AEK-POW-BMS63EN board versions

Table 9. AEK-POW-BMS63EN versions

Finished good	Schematic diagrams	Bill of materials
AEK$POW-BMS63ENA(1)	AEK$POW-BMS63ENA schematic diagrams	AEK$POW-BMS63ENA bill of materials

1. *This code identifies the AEK-POW-BMS63EN evaluation board first version. It is printed on the board PCB.*

16 AEK-POW-BMS63EN regulatory compliance information

Notice for US Federal Communication Commission (FCC)

For evaluation only; not FCC approved for resale

FCC NOTICE - This kit is designed to allow:

(1) Product developers to evaluate electronic components, circuitry, or software associated with the kit to determine

whether to incorporate such items in a finished product and

(2) Software developers to write software applications for use with the end product.

This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of this chapter, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of this chapter 3.1.2.

Notice for Innovation, Science and Economic Development Canada (ISED)

For evaluation purposes only. This kit generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to Industry Canada (IC) rules.

À des fins d'évaluation uniquement. Ce kit génère, utilise et peut émettre de l'énergie radiofréquence et n'a pas été testé pour sa conformité aux limites des appareils informatiques conformément aux règles d'Industrie Canada (IC).

Notice for the European Union

This device is in conformity with the essential requirements of the Directive 2014/30/EU (EMC) and of the Directive 2015/863/EU (RoHS).

Notice for the United Kingdom

This device is in compliance with the UK Electromagnetic Compatibility Regulations 2016 (UK S.I. 2016 No. 1091) and with the Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment Regulations 2012 (UK S.I. 2012 No. 3032).

17 AEK-POW-BMSNOTX schematic diagrams

Figure 61. **AEK-POW-BMSNOTX circuit schematic (1 of 2)**

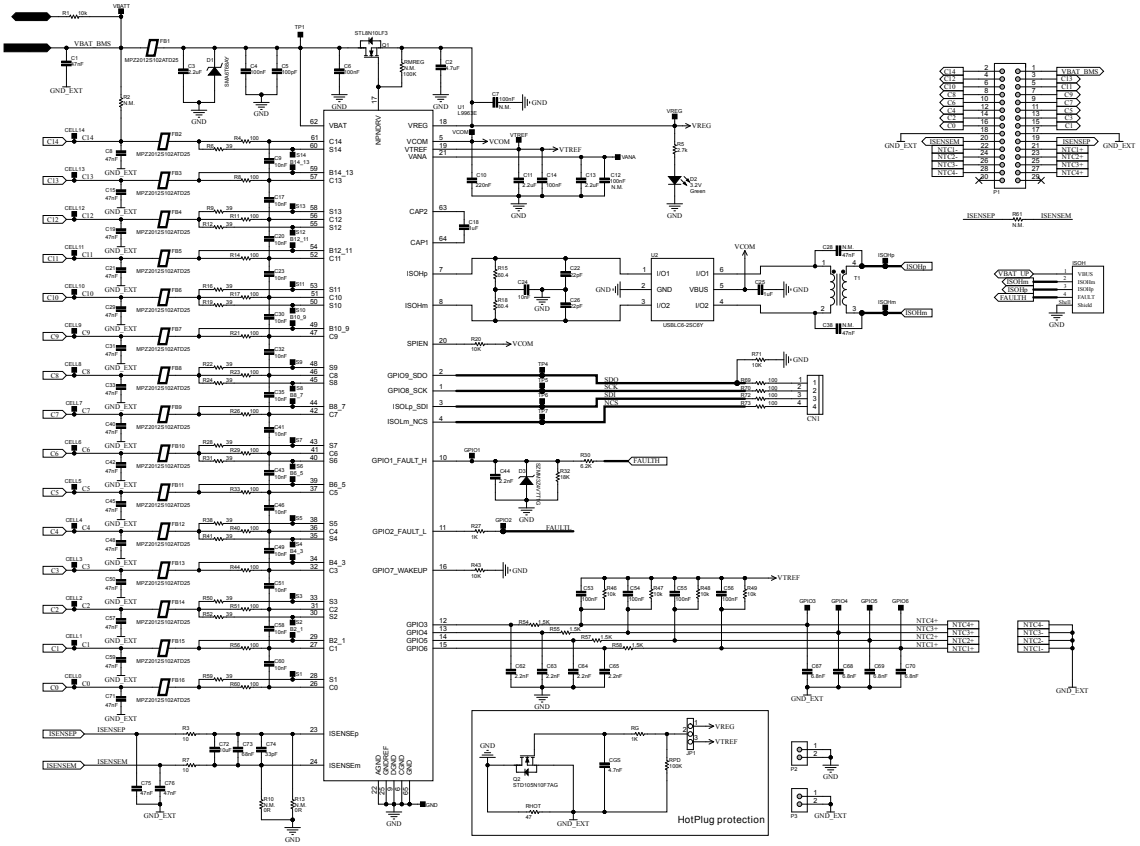
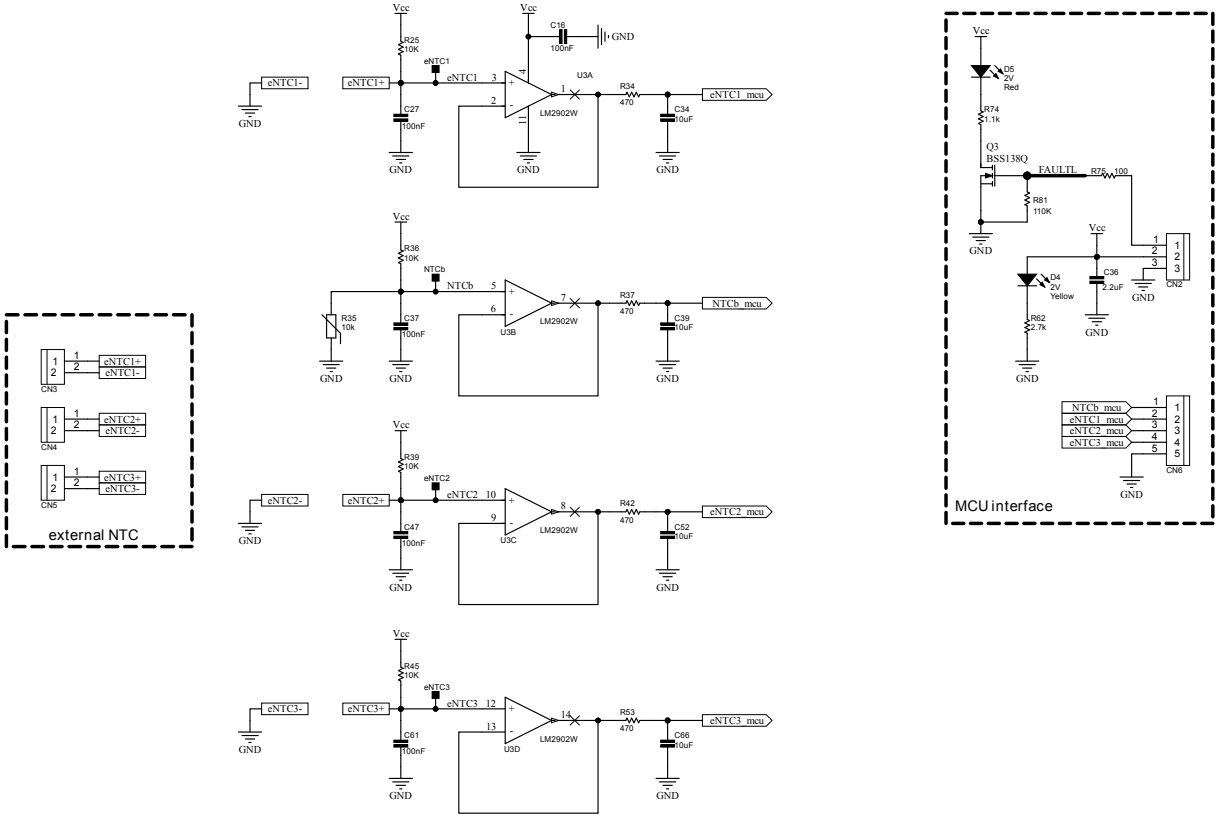


Figure 62. AEK-POW-BMSNOTX circuit schematic (2 of 2)



18 AEK-POW-BMSNOTX bill of materials

Table 10. **AEK-POW-BMSNOTX bill of materials**

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
1	18	C1, C8, C15, C19, C21, C29, C31, C33, C40, C42, C45, C48, C50, C57, C59, C71, C75, C76	47nF	0603 - 50V - X7R Class II	WE	885012206093
2	1	C2	4.7uF	1206 - 50V - X7R Class II	WE	885012208094
3	1	C3	2.2uF	1210 - 100V - X7R Class II	WE	885012209071
4	3	C4, C6, C14	100nF	0603 - 100V - X7R Class II	WE	885012206120
5	1	C5	100pF	0603 - 100V - X7R Class II	WE	885012206102
6	2	C7, C12	N.M.	0603	N.A.	N.A.
7	15	C9, C17, C20, C23, C24, C30, C32, C35, C41, C43, C46, C49, C51, C58, C60	10nF	0603 - 50V - X7R Class II	WE	885012206089
8	1	C10	220nF	0603 - 50V - X7R Class II	WE	885012206125
9	3	C11, C13, C36	2.2uF	0805 - 25V - X7R Class II	WE	885012207079
10	9	C16, C27, C37, C47, C53, C54, C55, C56, C61	100nF	0603 - 50V - X7R Class II	WE	885012206095
11	2	C18, C25	1uF	0805 - 50V - X7R Class II	WE	885012207103
12	2	C22, C26	22pF	0603 - 50V - NP0 Class I	WE	885012006053
13	2	C28, C38	N.M.	1206	N.A.	N.A.
14	4	C34, C39, C52, C66	10uF	0805 - 16V - X5R Class II	WE	885012107014
15	5	C44, C62, C63, C64, C65	2.2nF	0603 - 50V - X7R Class II	WE	885012206085
16	4	C67, C68, C69, C70	6.8nF	0603 - 50V - X7R Class II	WE	885012206088
17	1	C72	10uF	1210 - 50V - X7R Class II	WE	885012209073
18	1	C73	68nF	0603 - 50V - X7R Class II	WE	885012206094
19	1	C74	33pF	0603 - 50V - NP0 Class I	WE	885012006054
20	1	CGS	4.7nF	0603 - 50V - X7R Class II	WE	885012206087
21	1	CN1		2.54mm - 1 row - KK254 - Male	WE	61900411121
22	1	CN2		2.54mm - 1 row - KK254 - Male	WE	61900311121
23	3	CN3, CN4, CN5		2.54mm - 1 row - KK254 - Male	WE	61900211121
24	1	CN6		2.54mm - 1 row - KK254 - Male	WE	61900511121

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
25	1	D1	SMA6T68AY, SMA	Automotive 600 W, 68V TVS in SMA	ST	SMA6T68AY
26	1	D2	Green	0805 - Led Green - 3.2V	WE	150080GS75000
27	1	D3	SZMM3Z4V7T1G	4.7V Zener Voltage Regulators, 300mW	Onsemi	SZMM3Z4V7T1G
28	1	D4	Yellow	0805 - Led Yellow - 2V	WE	150080YS75000
29	1	D5	Red	0805 - Led Red - 2V	WE	150080RS75000
30	16	FB1, FB2, FB3, FB4, FB5, FB6, FB7, FB8, FB9, FB10, FB11, FB12, FB13, FB14, FB15, FB16	1K@100MHz	Ferrite Beads Multi-Layer Power 1KOhm 25% 100MHz 1.5A 0.15Ohm DCR 0805	TDK	MPZ2012S102ATD25
31	1	ISOH	61400416021	USB 2.0 Type A, Receptacle, Horizontal, THT	WE	61400416021
32	1	JP1		THT Vertical 3 pins Header, Pitch 2.54 mm, Single Row	WE	61300311121
33	1	P1		2.00mm - WR-WTB - Male Dual Row Horizontal Shrouded Header w. positive locking	WE	62403021722
34	2	P2, P3	61300211121	2.54mm - WR-PHD Pin Header, THT, pitch 2.54mm, Single Row, Vertical, 2p	WE	61300211121
35	1	Q1	STL8N10LF3, PowerFLAT 5x6 WF	Automotive-grade N-channel 100 V, 25 mΩ typ., 7.8 A STripFET™ F3 Power MOSFET in a PowerFLAT™ 5x6 package	ST	STL8N10LF3
36	1	Q2	STD105N10F7AG, DPAK	Automotive-grade N-channel 100 V, 6.8 mΩ typ., 80 A, STripFET™ F7 Power MOSFET in a DPAK package	ST	STD105N10F7AG
37	1	Q3	BSS138Q	N-Channel Enhancement Mosfet	NEXPERIA	BSS138Q-7-F
38	1	R1	10k	1206 - ±1% - 0.66W	Panasonic	ERJUP8F1002V
39	1	R2	N.M.	0805	N.A.	N.A.
40	2	R3, R7	10	0603 - ±1% - 0.25W	Panasonic	ERJPA3F10R0V
41	20	R4, R8, R11, R14, R17, R21, R23, R26, R29, R33, R40, R44, R51, R56, R60, R69, R70, R72, R73, R75	100	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1000V
42	2	R5, R62	2.7k	0603 - ±1% - 0.125W	Vishay	MCT06030C2701FP500
43	14	R6, R9, R12, R16, R19, R22, R24, R28, R31, R38, R41, R50, R52, R59	39	2010 - ±1% - 1.25W	TE Connectivity	CRGP2010F39R

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
44	3	R10, R13, RMREG	N.M.	0603	N.A.	N.A.
45	2	R15, R18	60.4	0603 - ±1% - 0.1W	Panasonic	ERJ3EKF60R4V
46	7	R20, R25, R36, R39, R43, R45, R71	10K	0603 - ±1% - 0.2W	Panasonic	ERJP03F1002V
47	2	R27, RG	1K	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1001V
48	1	R30	6.2K	0805 - ±1% - 0.5W	Panasonic	ERJP06F6201V
49	1	R32	18K	0603 - ±1% - 0.2W	Panasonic	ERJP03F1802V
50	4	R34, R37, R42, R53	470	0603 - ±1% - 0.125W	Panasonic	ERJH3EF4700V
51	1	R35	10k	0603 - ±1% - 0.1W	TDK	NTCG163JH103HTDS
52	4	R46, R47, R48, R49	10k	0805 - ±1% - 0.5W	Panasonic	ERJP6WF1002V
53	4	R54, R55, R57, R58	1.5K	2010 - ±1% - 2W	TE Connectivity	35021K5FT
54	1	R61	N.M.	2512	N.A.	N.A.
55	1	R74	1.1k	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1101V
56	1	R81	110K	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1103V
57	1	RHOT	47	2512 - ±5% - 1W	TE Connectivity	352047RJT
58	1	RPD	100K	0603 - ±1% - 0.25W	Panasonic	ERJP03F1003V
59	1	T1	120uH	Pulse Transformers 120uH	WE	74941000
60	1	U1	L9963E, TQFP 64 10x10x1.0	Automotive chip for battery management applications with daisy chain up to 31 devices	ST	L9963E
61	1	U2	USBLC6-2SC6Y, SOT23-6L	Automotive ESD protection for high speed interfaces.	ST	USBLC6-2SC6Y
62	1	U3	LM2902WYDT, SO-14	Low-power quad operational amplifier	ST	LM2902WYDT
63	1	for blister	60900213421	WR-PHD 2.54 mm Multi-Jumper Jumper with Test Point	WE	60900213421
64	4	for blister	970080365	WA-SPAII Plastic Spacer Stud, metric, internal/ internal	WE	970080365
65	4	for blister	97790603211	WA-SCRW Pan Head Screw w. cross slot M3	WE	97790603211
66	1	for blister	624030213322	WR-WTB 2.00 mm Female Dual Row Terminal Housing w. positive locking	WE	624030213322
67	30	for blister	62400113722	WR-WTB 2.00 mm Female Dual Row Crimp Contact	WE	62400113722
68	1	for blister	61900411621	WR-WTB 2.54 mm Female Terminal Housing	WE	61900411621
69	1	for blister	61900311621	WR-WTB 2.54 mm Female Terminal Housing	WE	61900311621
70	1	for blister	61900511621	WR-WTB 2.54 mm Female Terminal Housing	WE	61900511621
71	3	for blister	61900211621	WR-WTB 2.54 mm Female Terminal Housing	WE	61900211621

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
72	18	for blister	61910113722	WR-WTB 2.54 mm Female Crimp Contact	WE	61910113722

19 AEK-POW-BMSNOTX board versions

Table 11. AEK-POW-BMSNOTX versions

Finished good	Schematic diagrams	Bill of materials
AEK$POW-BMSNOTXA(1)	AEK$POW-BMSNOTXA schematic diagrams	AEK$POW-BMSNOTXA bill of materials

1. *This code identifies the AEK-POW-BMSNOTX evaluation board first version. It is printed on the board PCB.*

20 AEK-POW-BMSNOTX regulatory compliance information

Notice for US Federal Communication Commission (FCC)

For evaluation only; not FCC approved for resale

FCC NOTICE - This kit is designed to allow:

(1) Product developers to evaluate electronic components, circuitry, or software associated with the kit to determine

whether to incorporate such items in a finished product and

(2) Software developers to write software applications for use with the end product.

This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of this chapter, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of this chapter 3.1.2.

Notice for Innovation, Science and Economic Development Canada (ISED)

For evaluation purposes only. This kit generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to Industry Canada (IC) rules.

À des fins d'évaluation uniquement. Ce kit génère, utilise et peut émettre de l'énergie radiofréquence et n'a pas été testé pour sa conformité aux limites des appareils informatiques conformément aux règles d'Industrie Canada (IC).

Notice for the European Union

This device is in conformity with the essential requirements of the Directive 2014/30/EU (EMC) and of the Directive 2015/863/EU (RoHS).

Notice for the United Kingdom

This device is in compliance with the UK Electromagnetic Compatibility Regulations 2016 (UK S.I. 2016 No. 1091) and with the Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment Regulations 2012 (UK S.I. 2012 No. 3032).

21 AEK-POW-BMSWTX schematic diagrams

Figure 63. **AEK-POW-BMSWTX circuit schematic (1 of 2)**

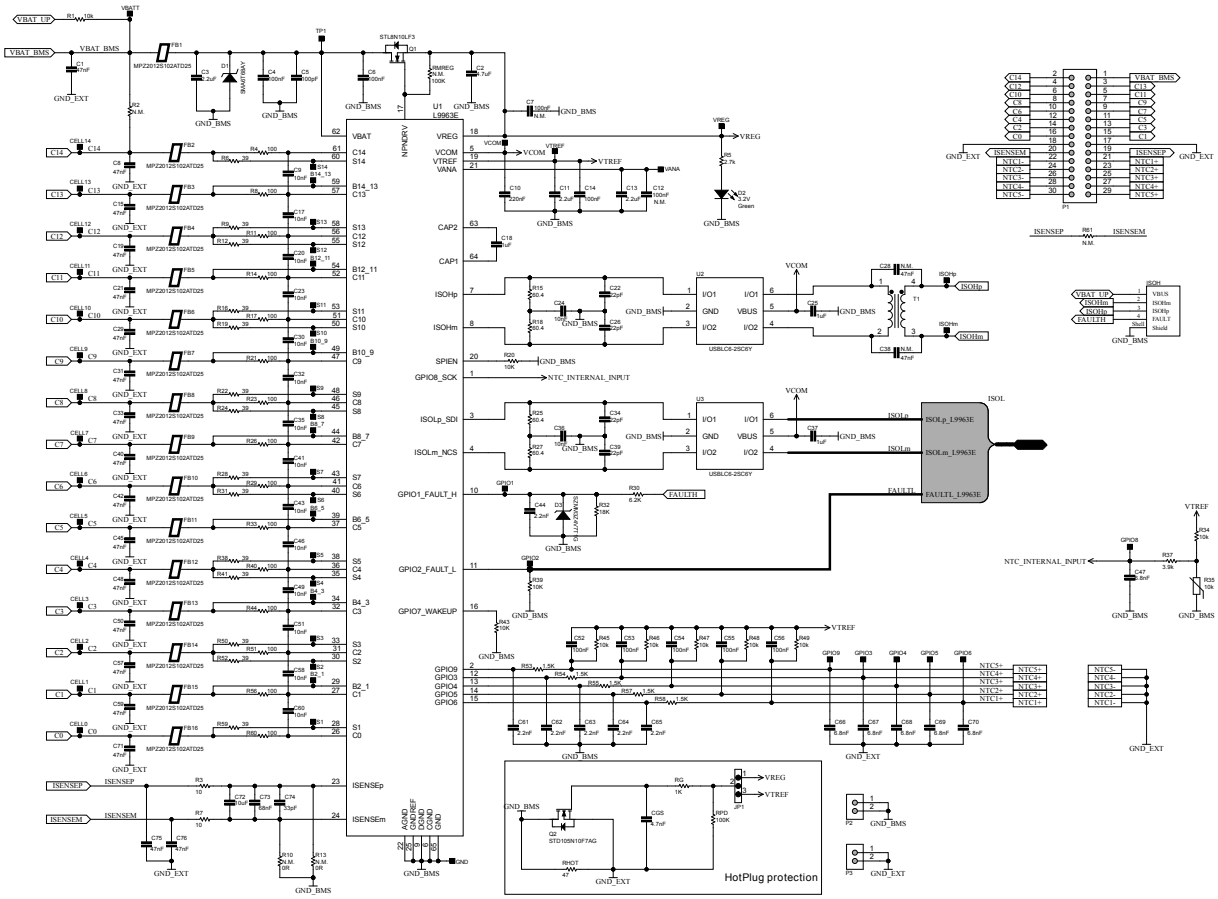
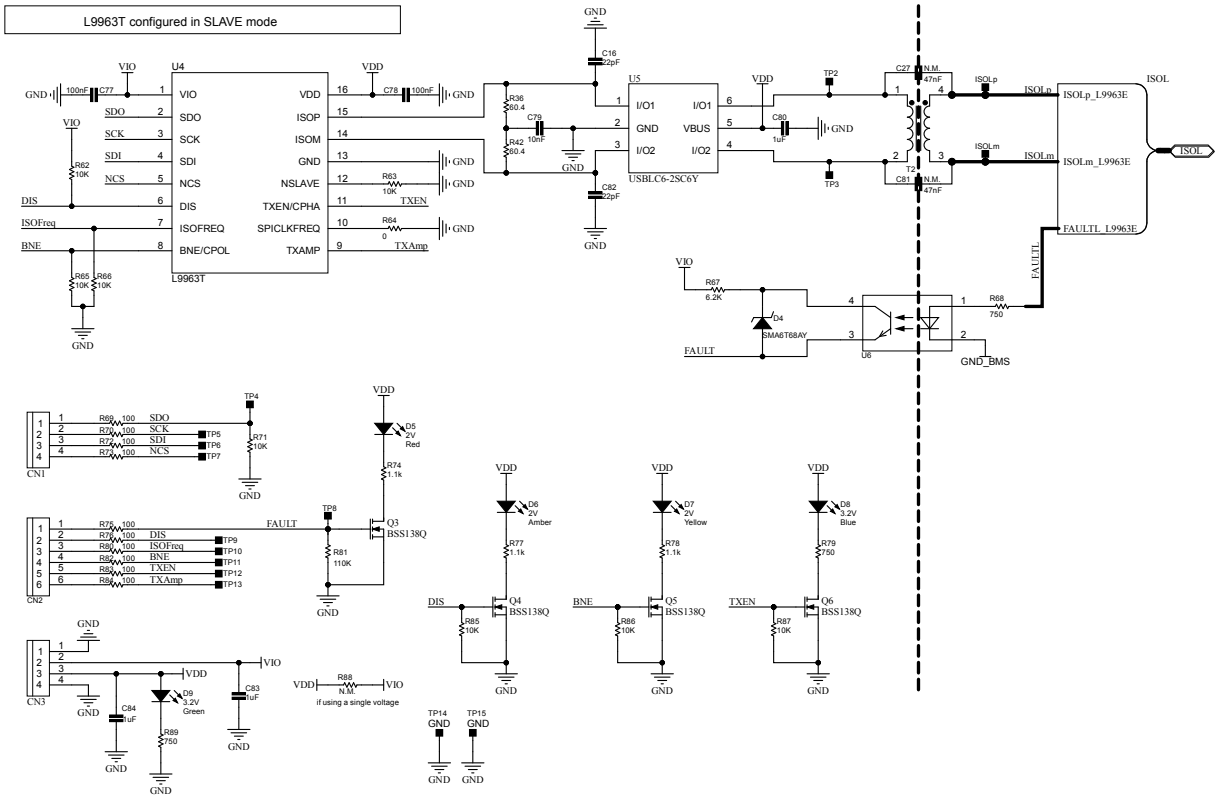


Figure 64. **AEK-POW-BMSWTX circuit schematic (2 of 2)**



22 AEK-POW-BMSWTX bill of materials

Table 12. **AEK-POW-BMSWTX bill of materials**

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
1	18	C1, C8, C15, C19, C21, C29, C31, C33, C40, C42, C45, C48, C50, C57, C59, C71, C75, C76	47nF	0603 - 50V - X7R Class II	WE	885012206093
2	1	C2	4.7uF	1206 - 50V - X7R Class II	WE	885012208094
3	1	C3	2.2uF	1210 - 100V - X7R Class II	WE	885012209071
4	3	C4, C6, C14	100nF	0603 - 100V - X7R Class II	WE	885012206120
5	1	C5	100pF	0603 - 100V - X7R Class II	WE	885012206102
6	2	C7, C12	N.M.	0603	N.A.	N.A.
7	17	C9, C17, C20, C23, C24, C30, C32, C35, C36, C41, C43, C46, C49, C51, C58, C60, C79	10nF	0603 - 50V - X7R Class II	WE	885012206089
8	1	C10	220nF	0603 - 50V - X7R Class II	WE	885012206125
9	2	C11, C13	2.2uF	0805 - 25V - X7R Class II	WE	885012207079
10	6	C16, C22, C26, C34, C39, C82	22pF	0603 - 50V - NP0 Class I	WE	885012006053
11	6	C18, C25, C37, C80, C83, C84	1uF	0805 - 50V - X7R Class II	WE	885012207103
12	4	C27, C28, C38, C81	N.M.	1206	N.A.	N.A.
13	6	C44, C61, C62, C63, C64, C65	2.2nF	0603 - 50V - X7R Class II	WE	885012206085
14	6	C47, C66, C67, C68, C69, C70	6.8nF	0603 - 50V - X7R Class II	WE	885012206088
15	7	C52, C53, C54, C55, C56, C77, C78	100nF	0603 - 50V - X7R Class II	WE	885012206095
16	1	C72	10uF	1210 - 50V - X7R Class II	WE	885012209073
17	1	C73	68nF	0603 - 50V - X7R Class II	WE	885012206094
18	1	C74	33pF	0603 - 50V - NP0 Class I	WE	885012006054
19	1	CGS	4.7nF	0603 - 50V - X7R Class II	WE	885012206087
20	2	CN1, CN3		2.54mm - 1 row - KK254 - Male	WE	61900411121

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
21	1	CN2		2.54mm - 1 row - KK254 - Male	WE	61900611121
22	2	D1, D4	SMA6T68AY, SMA	Automotive 600 W, 68V TVS in SMA	ST	SMA6T68AY
23	2	D2, D9	Green	0805 - Led Green - 3.2V	WE	150080GS75000
24	1	D3	SZMM3Z4V7T1G	4.7V Zener Voltage Regulators, 300mW	Onsemi	SZMM3Z4V7T1G
25	1	D5	Red	0805 - Led Red - 2V	WE	150080RS75000
26	1	D6	Amber	0805 - Led Amber - 2V	WE	150080AS75000
27	1	D7	Yellow	0805 - Led Yellow - 2V	WE	150080YS75000
28	1	D8	Blue	0805 - Led Blue - 3.2V	WE	150080BS75000
29	16	FB1, FB2, FB3, FB4, FB5, FB6, FB7, FB8, FB9, FB10, FB11, FB12, FB13, FB14, FB15, FB16	1K@100MHz	Ferrite Beads Multi-Layer Power 1KOhm 25% 100MHz 1.5A 0.15Ohm DCR 0805	TDK	MPZ2012S102ATD25
30	1	ISOH	61400416021	USB 2.0 Type A, Receptacle, Horizontal, THT	WE	61400416021
31	1	JP1		THT Vertical 3 pins Header, Pitch 2.54 mm, Single Row	WE	61300311121
32	1	P1		2.00mm - WR-WTB - Male Dual Row Horizontal Shrouded Header w. positive locking	WE	62403021722
33	2	P2, P3	61300211121	2.54mm - WR-PHD Pin Header, THT, pitch 2.54mm, Single Row, Vertical, 2p	WE	61300211121
34	1	Q1	STL8N10LF3, PowerFLAT 5x6 WF	Automotive-grade N-channel 100 V, 25 mΩ typ., 7.8 A STripFET™ F3 Power MOSFET in a PowerFLAT™ 5x6 package	ST	STL8N10LF3

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
35	1	Q2	STD100N10F7, DPAK	N-channel 100 V, 6.8 mΩ typ., 80 A STripFET F7 Power MOSFETs in D2PAK, DPAK, TO-220FP, I2PAK and TO-220 packages STripFET™ F7 Power MOSFET in a DPAK package	ST	STD100N10F7
36	4	Q3, Q4, Q5, Q6	BSS138Q	N-Channel Enhancement Mosfet	NEXPERIA	BSS138Q-7-F
37	1	R1	10k	1206 - ±1% - 0.66W	Panasonic	ERJUP8F1002V
38	2	R2, R88	N.M.	0805	N.A.	N.A.
39	2	R3, R7	10	0603 - ±1% - 0.25W	Panasonic	ERJPA3F10R0V
40	25	R4, R8, R11, R14, R17, R21, R23, R26, R29, R33, R40, R44, R51, R56, R60, R69, R70, R72, R73, R75, R76, R80, R82, R83, R84	100	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1000V
41	1	R5	2.7k	0603 - ±1% - 0.125W	Vishay	MCT06030C2701FP500
42	14	R6, R9, R12, R16, R19, R22, R24, R28, R31, R38, R41, R50, R52, R59	39	2010 - ±1% - 1.25W	TE Connectivity	CRGP2010F39R
43	3	R10, R13, RMREG	N.M.	0603	N.A.	N.A.
44	6	R15, R18, R25, R27, R36, R42	60.4	0603 - ±1% - 0.1W	Panasonic	ERJ3EKF60R4V
45	11	R20, R39, R43, R62, R63, R65, R66, R71, R85, R86, R87	10K	0603 - ±1% - 0.2W	Panasonic	ERJP03F1002V
46	2	R30, R67	6.2K	0805 - ±1% - 0.5W	Panasonic	ERJP06F6201V
47	1	R32	18K	0603 - ±1% - 0.2W	Panasonic	ERJP03F1802V
48	6	R34, R45, R46, R47, R48, R49	10k	0805 - ±1% - 0.5W	Panasonic	ERJP6WF1002V
49	1	R35	10k	0603 - ±1% - 0.1W	TDK	NTCG163JH103HTDS
50	1	R37	3.9k	0603 - ±1% - 0.1W	Panasonic	ERJ3EKF3901V

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
51	5	R53, R54, R55, R57, R58	1.5K	2010 - ±1% - 2W	TE Connectivity	35021K5FT
52	1	R61	N.M.	N.A.	N.A.	N.A.
53	1	R64	0	0603 - ±1% - 0.1W	Panasonic	ERJ3GEY0R00V
54	3	R68, R79, R89	750	0603 - ±0.5% - 0.25W, 0603 - ±1% - 0.25W	Panasonic	ERJUP3D7500V
55	3	R74, R77, R78	1.1k	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1101V
56	1	R81	110K	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1103V
57	1	RG	1K	0603 - ±1% - 0.25W	Panasonic	ERJPA3F1001V
58	1	RHOT	47	2512 - ±5% - 1W	TE Connectivity	352047RJT
59	1	RPD	100K	0603 - ±1% - 0.25W	Panasonic	ERJP03F1003V
60	2	T1, T2	125uH	Pulse Transformers 125uH	WE	74941000
61	1	U1	L9963E, TQFP 64 10x10x1.0	Automotive chip for battery management applications with daisy chain up to 31 devices	ST	L9963E
62	3	U2, U3, U5	USBLC6-2SC6Y, SOT23-6L	Automotive ESD protection for high speed interfaces.	ST	USBLC6-2SC6Y
63	1	U4	L9963T, SO-16	Automotive general purpose SPI to isolated SPI transceiver	ST	L9963T
64	1	U6	140357145300	WL-OCPT Optocoupler Phototransistor, SOP4, 1 Channel, DC, 35V, 60mA	WE	140357145300
65	1	for blister	60900213421	WR-PHD 2.54 mm Multi-Jumper Jumper with Test Point	WE	60900213421
66	4	for blister	970080365	WA-SPAII Plastic Spacer Stud, metric, internal/ internal	WE	970080365
67	4	for blister	97790603211	WA-SCRW Pan Head Screw w. cross slot M3	WE	97790603211

Item	Q.ty	Ref.	Part/value	Description	Manufacturer	Order code
68	1	for blister	624030213322	WR-WTB 2.00 mm Female Dual Row Terminal Housing w. positive locking	WE	624030213322
69	30	for blister	62400113722	WR-WTB 2.00 mm Female Dual Row Crimp Contact	WE	62400113722
70	2	for blister	61900411621	WR-WTB 2.54 mm Female Terminal Housing	WE	61900411621
71	1	for blister	61900611621	WR-WTB 2.54 mm Female Terminal Housing	WE	61900611621
72	14	for blister	61910113722	WR-WTB 2.54 mm Female Crimp Contact	WE	61910113722

23 AEK-POW-BMSWTX Board versions

Table 13. AEK-POW-BMSWTX versions

Finished good	Schematic diagrams	Bill of materials
AEK$POW-BMSWTXA(1)	AEK$POW-BMSWTXA schematic diagrams	AEK$POW-BMSWTXA bill of materials

1. *This code identifies the AEK-POW-BMSWTX evaluation board first version. It is printed on the board PCB.*

24 AEK-POW-BMSWTX regulatory compliance information

Notice for US Federal Communication Commission (FCC)

For evaluation only; not FCC approved for resale

FCC NOTICE - This kit is designed to allow:

(1) Product developers to evaluate electronic components, circuitry, or software associated with the kit to determine

whether to incorporate such items in a finished product and

(2) Software developers to write software applications for use with the end product.

This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of this chapter, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of this chapter 3.1.2.

Notice for Innovation, Science and Economic Development Canada (ISED)

For evaluation purposes only. This kit generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to Industry Canada (IC) rules.

À des fins d'évaluation uniquement. Ce kit génère, utilise et peut émettre de l'énergie radiofréquence et n'a pas été testé pour sa conformité aux limites des appareils informatiques conformément aux règles d'Industrie Canada (IC).

Notice for the European Union

This device is in conformity with the essential requirements of the Directive 2014/30/EU (EMC) and of the Directive 2015/863/EU (RoHS). Compliance to EMC standards in Class A (industrial intended use).

Notice for the United Kingdom

This device is in compliance with the UK Electromagnetic Compatibility Regulations 2016 (UK S.I. 2016 No. 1091) and with the Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment Regulations 2012 (UK S.I. 2012 No. 3032). Compliance to EMC standards in Class A (industrial intended use).

Revision history

Table 14. **Document revision history**

Date	Version	Changes
16-May-2023	1	Initial release.
20-May-2024	2	Updated title in cover page. Added Section 1.2: AEK-POW-BMSNOTX overview Minor text changes.
08-Aug-2024	3	Updated Section Introduction, Section 1.5.2.2.2: Addressing procedure, Section 2: BMS topologies, Section 3: Voltage conversion routine, Section 7: Fault condition in daisy chains, Section 10: AutoDevKit ecosystem , Section 10.5: Available demos, Section 11: Available APIs and Section 12.1: Process timing. Added Section 1: BMS evaluation board overview, Section 1.4: Embedded devices, Section 5: Cell current measurement, Section 21: AEK-POW-BMSWTX schematic diagrams, Section 22: AEK-POW-BMSWTX bill of materials, Section 23: AEK-POW-BMSWTX Board versions and Section 24: AEK-POW-BMSWTX regulatory compliance information. Minor text changes.

Contents

List of figures

List of tables